REFORMA 2017
WYDANIE ZMIENIONE

INFORMATYKA

podręcznik

Oznaczono zmiany i wykreślenia

E-book na eduranga.pl

PODSTAWA 2024





8

szkoła podstawowa

Drogi WSiP,

Nigdy bym nie publikowała publicznie książek wydawnictw, które działają na uczciwych zasadach.

Wasza firma jednak promuje masowy dodruk, całkowicie niepotrzebnych książek, które mogłyby być zastąpione wersjami elektronicznymi!

Co prawda e-booki są dostępne na waszej stronie, jednak:

- W przeciwieństwie do fizycznej książki, licencja na e-book kończy się po roku. Oznacza to,
- że jeżeli moja córka chciałaby powtórzyć sobie całą wiedzę do matury, musiałabym jej kupić wszystkie wasze książki od nowa.
- Waszych e-booków nie da się pobrać! Wymagają one dostęp do internetu, co uniemożliwia ich użycie na naszej wsi, gdzie zasięg jest ograniczony.
- Wasze e-booki nie działają na telefonach komórkowych!
- Wasze e-booki sprzedawane są po tej samej (albo wyższej) cenie co regularne książki. Cena e-booka powinna być niższa, gdyż e-booki wymagają elektronicznego czytnika (tabletu)!

Czas rozpocząć nową erę, w której papier nie jest bezczelnie marnowany dla pieniędzy. Przedstawiam e-book, który spełnia wszystkie oczekiwania uczniów.

Dbajmy o środowisko, zróbmy to dla młodych pokoleń.

SPIS TREŚCI

 treści usunięte wg podstawy programowej 2024

 treści fakultatywne wg podstawy programowej 2024

O podręczniku

Strona działowa

Krótkie wprowadzenie w tematykę rozdziału

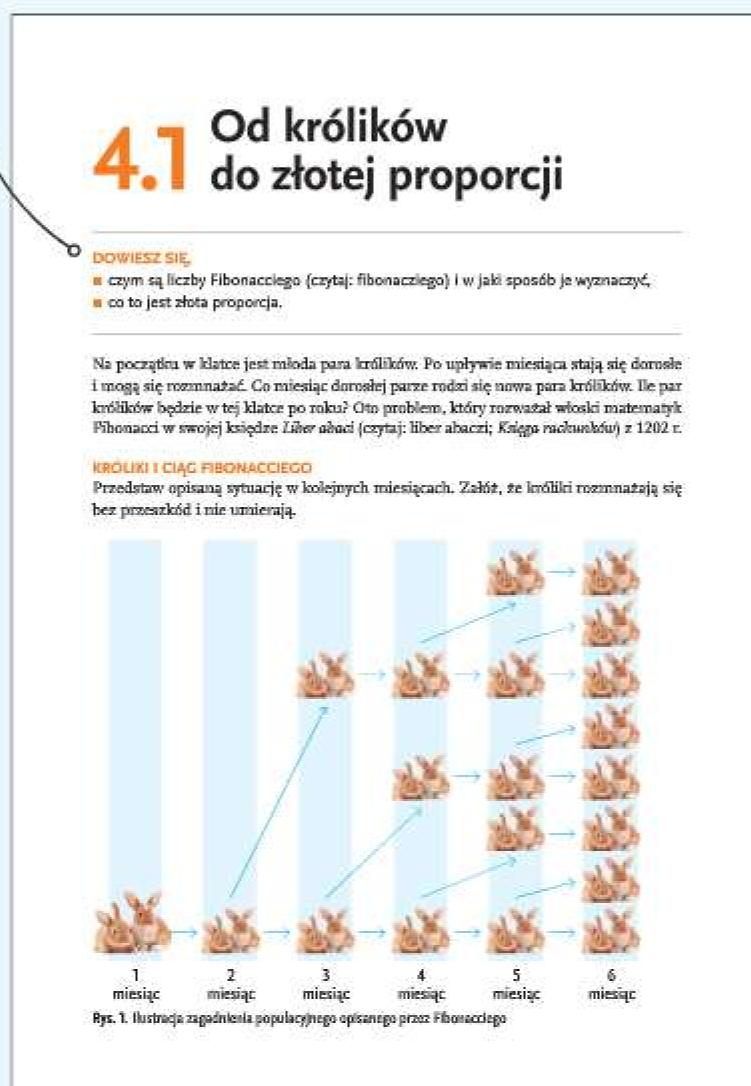


1 Lekcje z HTML-em

Idea World Wide Web (WWW, W3) powstała w 1989 r. Jej pomysłodawcą był Tim Berners-Lee (czytaj: berners-li), brytyjski fizyk i programista. Wraz z belgijskim programistą Robertem Cailliau (czytaj: kaju) stworzyli obsługiwany za pomocą przeglądarki internetowej system, którego działanie opiera się na sieci odnośników – hiperłączy. Berners-Lee opublikował pierwszą stronę WWW 20 grudnia 1990 r. w sieci Europejskiego Ośrodka Badań Jądrowych (CERN) w Genewie. Przedstawiała ona szczegóły projektu. Pod adresem **info.cern.ch/hypertext/WWW/TheProject.html** można obejrzeć kopię strony z 1992 r.

Lista zagadnień

Omawiane tematy i kluczowe umiejętności



4.1 Od królików do złotej proporcji

DOWIESZ SIĘ,
- czym są liczby Fibonacciego (czytaj: fibonaczjego) i w jaki sposób je wyznaczyć,
- co to jest złota proporcja.

Na początku w klatce jest młoda para królików. Po upływie miesiąca stają się dorosłe i mogą się rozmnażać. Co miesiąc dorosłej parze rodzi się nowa para królików. Ile par królików będzie w tej klatce po roku? Oto problem, który rozważał włoski matematyk Fibonacci w swojej księdze *Liber abaci* (czytaj: liber abaczi; *Księga rachunków*) z 1202 r.

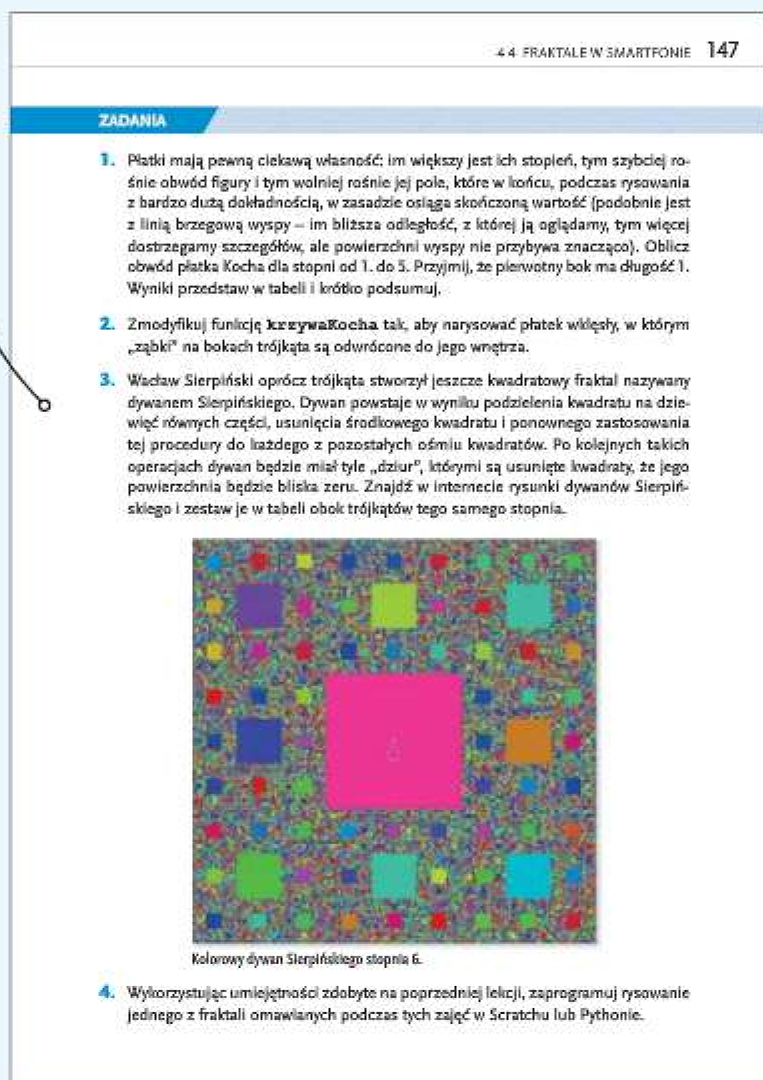
KRÓLIKI I CIĄG FIBONACCIEGO

Przedstaw opisaną sytuację w kolejnych miesiącach. Załóż, że króliki rozmnażają się bez przeszkód i nie umierają.

Rys. 1. Ilustracja zagadnienia populacyjnego opisanego przez Fibonacciego

Zadania

Pomoc w utrwalaniu umiejętności zdobytych na lekcji



4.4 FRAKTALE W SMARTFONIE 147

ZADANIA

1. Płatki mają pewną ciekawą własność: im większy jest ich stopień, tym szybciej rośnie obwód figury i tym wolniej rośnie jej pole, które w końcu, podczas rysowania z bardzo dużą dokładnością, w zasadzie osiąga skończoną wartość (podobnie jest z linią brzegową wyspy – im bliższa odległość, z której ją oglądamy, tym więcej dostrzegamy szczegółów, ale powierzchni wyspy nie przybywa znacząco). Oblicz obwód płatka Kocha dla stopni od 1. do 5. Przyjmij, że pierwotny bok ma długość 1. Wyniki przedstaw w tabeli i krótko podsumuj.
2. Zmodyfikuj funkcję `krzywaKocha` tak, aby narysować płatek wklęsły, w którym „ząbki" na bokach trójkąta są odwrócone do jego wnętrza.
3. Wacław Sierpiński oprócz trójkąta stworzył jeszcze kwadratowy fraktal nazywany dywanem Sierpińskiego. Dywan powstaje w wyniku podzielenia kwadratu na dziewięć równych części, usunięcia środkowego kwadratu i ponownego zastosowania tej procedury do każdego z pozostałych ośmiu kwadratów. Po kolejnych takich operacjach dywan będzie miał tyle „dziur", którymi są usunięte kwadraty, że jego powierzchnia będzie bliska zeru. Znajdź w internecie rysunki dywanów Sierpińskiego i zestaw je w tabeli obok trójkątów tego samego stopnia.

Kolorowy dywan Sierpińskiego stopnia 6.

4. Wykorzystując umiejętności zdobyte na poprzedniej lekcji, zaprogramuj rysowanie jednego z fraktali omawianych podczas tych zajęć w Scratchu lub Pythonie.

Podsumowanie

Skrót najważniejszych wiadomości z danego rozdziału



84 PODSUMOWANIE

Jak zapisać definicje funkcji w Pythonie?

Definicję funkcji rozpoczyna słowo `def`, następnie jest nazwa funkcji, a za nią nawias z parametrami lub pusty. Kolejne wiersze to polecenia do wykonania. Aby wywołać funkcję, wystarczy podać jej nazwę oraz nawias z wartością parametru lub pusty.

Czym różnią się pętle while i for?

Iterację, czyli powtarzanie ciągu poleceń w pętli określoną liczbę razy lub aż do spełnienia określonego warunku, można wykonać odpowiednio za pomocą pętli `for` oraz `while`. Pętla `for` wykonuje ciąg poleceń tyle razy, ile wynosi parametr funkcji `range()`.

```
for i in range(3):
  fd(100); lt(120)
```

Pętla `while` wykonuje ciąg poleceń, dopóki podany warunek jest prawdziwy.

```
a = -10
while a < 0:
  a += 1
```

Czym różni się wyszukiwanie liniowe od wyszukiwania binarnego?

Wyszukiwanie liniowe polega na przeglądaniu kolejnych elementów i wymaga w najgorszym przypadku przejrzenia wszystkich elementów zbioru. Wyszukiwanie binarne opiera się na metodzie „dziel i zwyciężaj" – jeśli zatem uporządkowany zbiór danych ma milion elementów, to wystarczy sprawdzić maksymalnie 20 elementów w celu znalezienia żądanej wartości.

Pamiętaj!
Informatyka to podręcznik wieloletniego użytku, dlatego nie pisz po nim, bo będą z niego korzystali inni uczniowie.

1 Lekcje z HTML-em





Idea World Wide Web (WWW, W3) powstała w 1989 r. Jej pomysłodawcą był Tim Berners-Lee (czytaj: berners-li), brytyjski fizyk i programista. Wraz z belgijskim programistą Robertem Cailliau (czytaj: kaju) stworzyli obsługiwany za pomocą przeglądarki internetowej system, którego działanie opiera się na sieci odnośników – hiperłączy. Berners-Lee opublikował pierwszą stronę WWW 20 grudnia 1990 r. w sieci Europejskiego Ośrodka Badań Jądrowych (CERN) w Genewie. Przedstawiała ona szczegóły projektu. Pod adresem **info.cern.ch/hypertext/WWW/TheProject.html** można obejrzeć kopię strony z 1992 r.

STRUKTURA DOKUMENTU HTML

Stosowanie wcięć poprawia czytelność kodu i umożliwia szybsze znajdowanie ewentualnych błędów. W razie potrzeby kod można opatrzyć niewidocznymi docelowo dodatkowymi komentarzami, które ułatwią zrozumienie struktury albo usprawnią dalszą pracę przy projekcie.

dokument_HTML.htm - Notepad2

Plik Edycja Widok Ustawienia Pomoc

```
<!DOCTYPE HTML>

<html>

<!-- Strona WWW składa się z dwóch głównych części, tj. HEAD i BODY. Sekcja HEAD
odpowiada za opis strony. Sekcja BODY odpowiada za zawartość strony -->

  <head>

    <meta charset="UTF-8">
    <meta name="description" content="Strona opisuje strukturę dokumentu HTML">
    <meta name="keywords" content="struktura, head, body, znaczniki">
    <title>Struktura dokumentu HTML</title>
  </head>

  <body>

    <p>Na stronie wyświetlają się elementy zawarte w sekcji BODY, np. teksty,
obrazy, filmy, animacje, tabele, odnośniki, formularze.</p>

    </body>

</html>
```

Wiersz 21 : 21 Kolumna 8 Zaznaczono 0 643 B UTF-8 CR+LF INS Strona internetowa

dokument HTML zawierający strukturę strony internetowej w formie pliku tekstowego

Struktura dokumentu HTML

Plik | D:/dokument_HTML.htm

Na stronie wyświetlają się elementy zawarte w sekcji BODY, np. teksty, obrazy, filmy, animacje, tabele, odnośniki, formularze.

dokument HTML zinterpretowany przez przeglądarkę

Internauci korzystają z różnych systemów operacyjnych i przeglądarek, dlatego podstawową zasadą przy tworzeniu stron internetowych jest opracowanie takiej struktury dokumentu HTML, aby wyświetlał się poprawnie bez względu na preferencje użytkowników.

<!DOCTYPE HTML>
Informacja dla przeglądarki, że dokument, który otwiera, został napisany z zastosowaniem języka HTML 5.

<html>
Znacznik rozpoczynający dokument HTML.

<!-- -->
Komentarz widoczny dla osób przeglądających kod i pomijany przez przeglądarkę przy wyświetlaniu treści.

<head>
Początek sekcji nagłówkowej, w której umieszcza się znaczniki informacyjne dla przeglądarki oraz wyszukiwarki.

<meta charset="...">
Informacja o standardzie kodowania znaków. Zapis `charset="UTF-8"` gwarantuje, że przeglądarka będzie poprawnie interpretować polskie znaki.

<meta name="description" content="...">
Informacja dla wyszukiwarki zawierająca krótki opis strony.

<meta name="keywords" content="...">
Informacja dla wyszukiwarki zawierająca oddzielone przecinkami słowa kluczowe związane z tematem danej strony.

<title> ... </title>
Informacja dla wyszukiwarki zawierająca tytuł strony widoczny na pasku przeglądarki internetowej. Tytuł powinien być zwięzły i odnosić się do zawartości strony/serwisu.

</head>
Koniec sekcji nagłówkowej dokumentu.

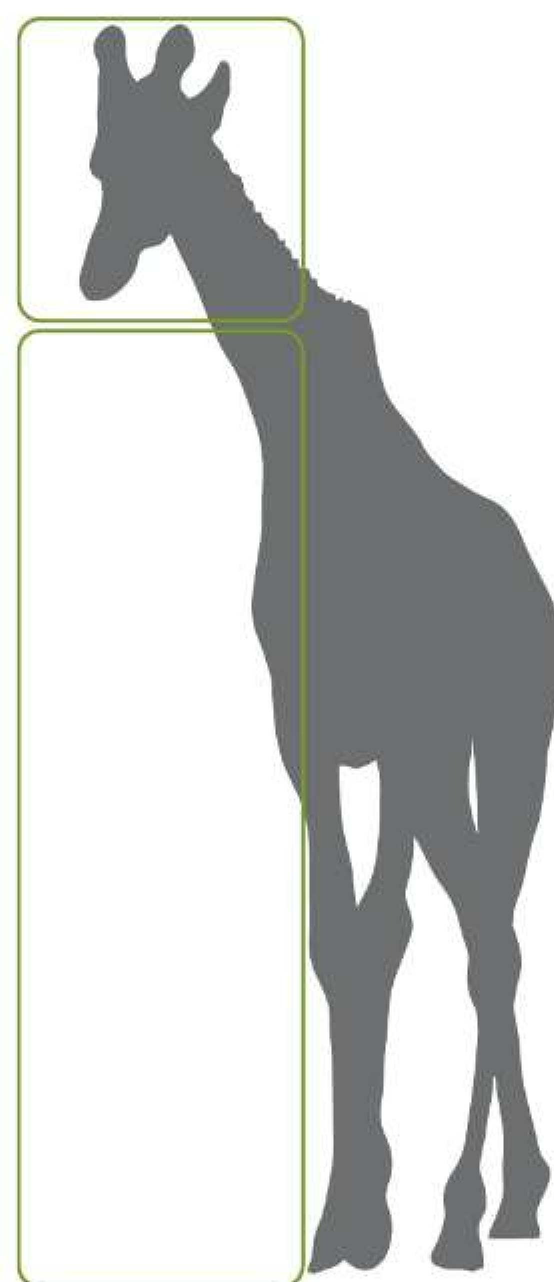
<body>
Początek właściwej treści dokumentu, która zostanie bezpośrednio wyświetlona w oknie przeglądarki. Oprócz tekstu może zawierać zdjęcia, tabele, filmy, linki.

<p>
Początek akapitu.

</p>
Koniec akapitu.

</body>
Koniec treści dokumentu.

</html>
Koniec dokumentu HTML.



1.1 Jak to zrobić w HTML-u i CSS?

DOWIESZ SIĘ,

- jakich programów używa się do tworzenia stron internetowych,
- jaka jest struktura dokumentu HTML,
- jak definiować style w dokumencie HTML.

Na pierwszy rzut oka tworzenie stron internetowych może wydawać się trudne, ale za chwilę przekonasz się, że to nieprawda. Najważniejsze, żeby przestrzegać zasad projektowania oraz odpowiednio ułożyć wszystkie elementy. I nie potrzebujesz skomplikowanych programów – wystarczy najprostszy edytor tekstu!

PROSTA STRONA WWW

Strona internetowa to dokument tekstowy opisany z użyciem instrukcji, które umożliwiają odpowiednie wyświetlenie zawartej w nim treści, udostępniony w sieci na serwerze WWW i otwierany za pomocą przeglądarki.

Nieskomplikowane dokumenty tego typu można tworzyć za pomocą prostych edytorów tekstu, takich jak Notatnik dostępny w systemie Windows czy rozpowszechniany na zasadzie wolnej licencji Notepad2 (czytaj: nołtpadtu). Notepad2 jest nieco wygodniejszy od Notatnika, ponieważ zawiera mechanizmy numerowania wierszy, sygnalizowania błędów w zapisie i automatycznego wyróżniania instrukcji w kodzie źródłowym. Program można pobrać z serwisu autora polskiej wersji programu, Krzysztofa Knapika – **knapsu.eu/notepad2_pl**.

► Napisz w jednym ze wspomnianych powyżej edytorów notkę zatytułowaną *Cała ja* lub *Cały ja*. Sformatuj tekst, nadając mu elegancki wygląd.

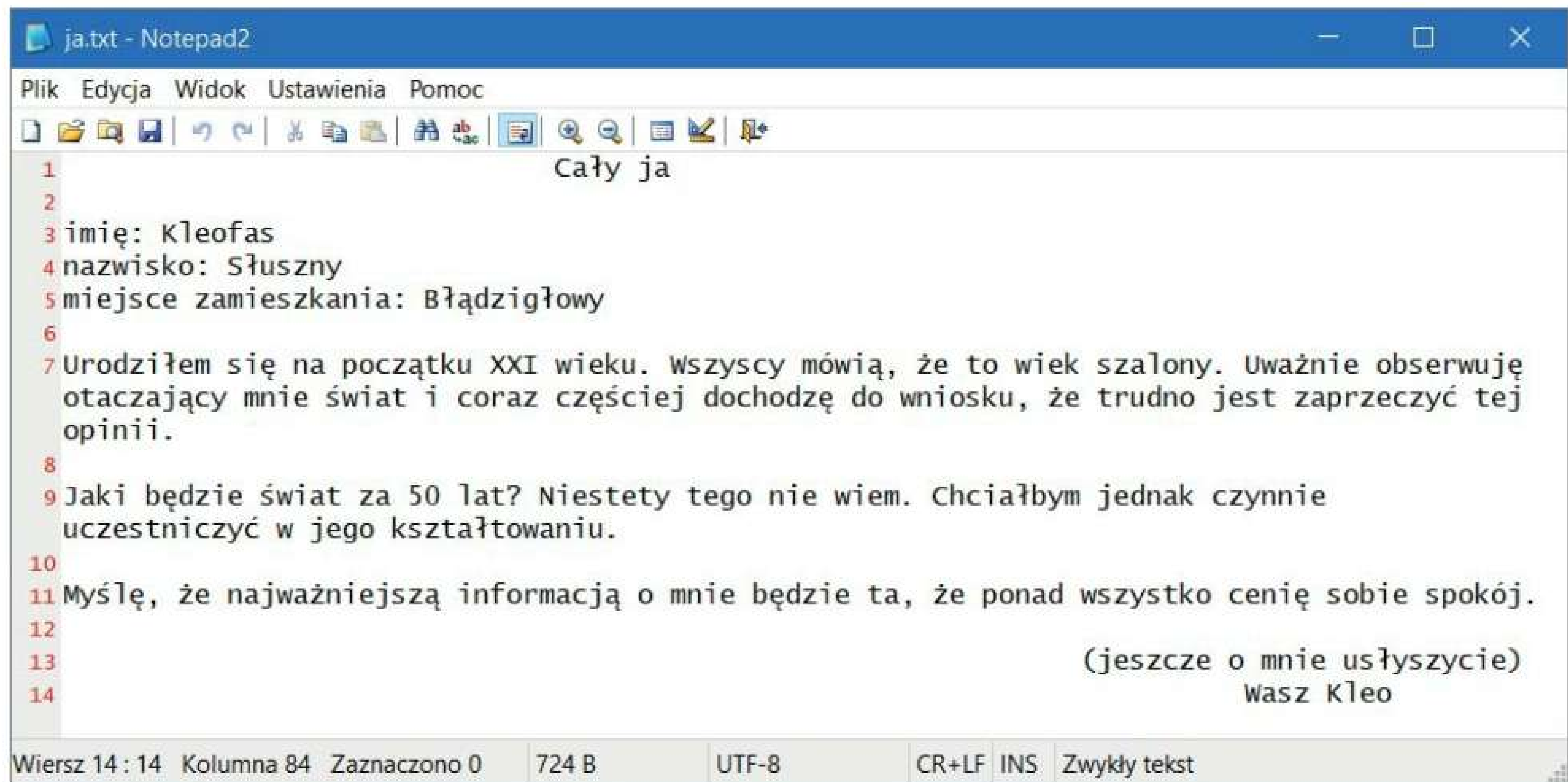
► Wprowadź ustawienia dotyczące kodowania, tak aby uniknąć problemów z wyświetlaniem polskich znaków diakrytycznych:

- w programie Notatnik wybierz **Zapisz jako...** → **Kodowanie** → **UTF-8**;
- w programie Notepad2 kliknij dwukrotnie pozycję **ANSI** na pasku stanu, a następnie w oknie **Kodowanie** w polu **Wybierz aktualne** zaznacz pozycję **Unicode (UTF-8)** i zatwierdź ustawienie przyciskiem **OK**.

► Zapisz notatkę dwukrotnie – nadaj plikom nazwy **ja.txt** oraz **ja.htm**.

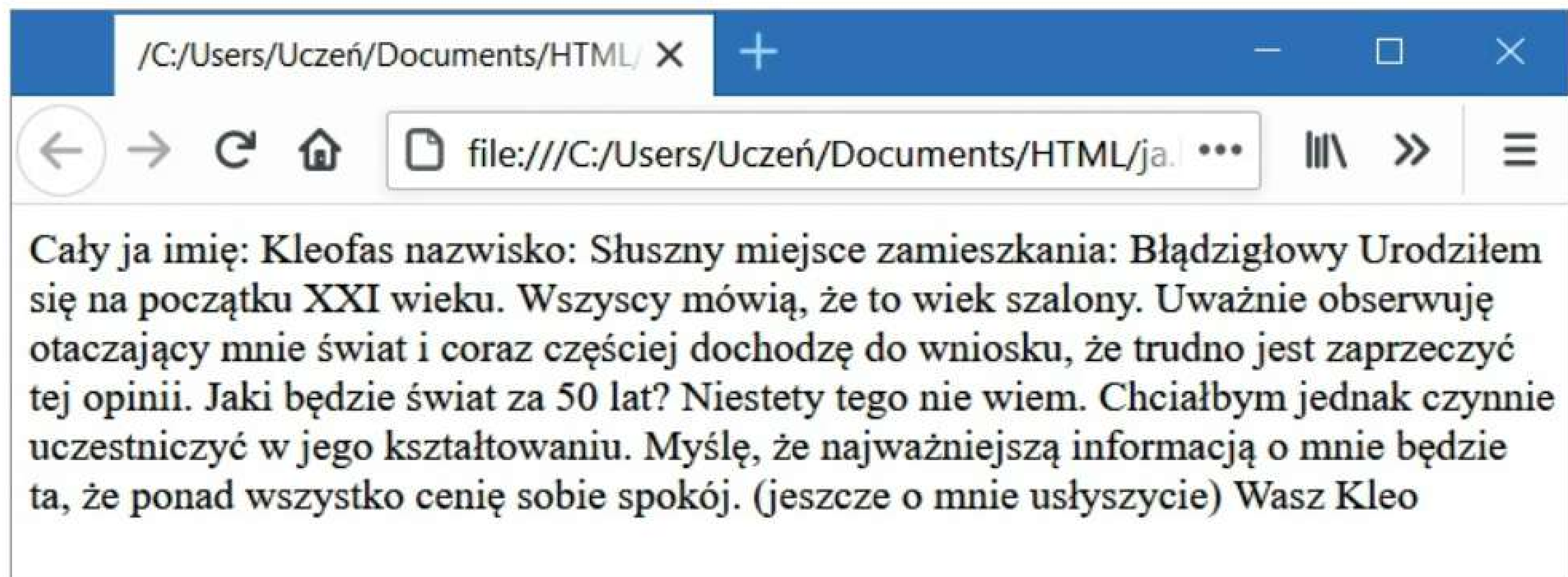
► Zamknij okno programu.

- A teraz otwórz i porównaj oba pliki.
  - Dokument **ja.txt** otwarty w dowolnym prostym edytorze tekstu wygląda dokładnie tak, jak powinien.



Rys. 1. Przykładowy tekst otwarty w programie Notepad2

  - Dokument **ja.htm** otwarty w domyślnej przeglądarce nie do końca przypomina pierwotnie sformatowany dokument, ponieważ przeglądarki ignorują formatowanie, które nie zostało zdefiniowane w odpowiedni sposób.



Rys. 2. Przykładowy tekst otwarty w oknie przeglądarki

STRUKTURA DOKUMENTU HTML

Podstawowa struktura dokumentu HTML została zdefiniowana przez World Wide Web Consortium (czytaj: łerld łajd łeb konsortiem; w skrócie W3C) – międzynarodowe forum złożone z firm, agencji rządowych oraz organizacji, które zajmuje się ustanawianiem standardów dotyczących stron WWW.

Do opisu struktury dokumentu służy HyperText Markup Language (czytaj: hajpertekst markap lengłidż) – język znaczników hipertekstu, znany powszechnie pod nazwą HTML. Jest on ciągle modyfikowany.

Znaczniki

Znaczniki (nazywane również tagami) to umieszczone między znakami **< >** polecenia dla przeglądarki opisujące sposób wyświetlania strony. Zwykle stosuje się pary znaczników – otwierający i zamykający, np. początek nagłówka **<h1>** i koniec nagłówka **</h1>**, ale istnieją też znaczniki pojedyncze, np. znacznik **
**, który odpowiada za przejście do następnego wiersza tekstu.

- Przeanalizuj infografikę *Struktura dokumentu HTML* – zwróć uwagę zarówno na znaczniki, jak i na sposób zapisu (m.in. wcięcia, wyraźne oddzielenie znaczników od treści oraz komentarze dotyczące struktury).
- Zmodyfikuj dokument **ja.htm** – dodaj znaczniki, tak aby dokument zyskał poprawną strukturę. Następnie zapisz go pod nazwą **ja1.htm**.
  - Aby otworzyć dokument do edycji, kliknij ikonę pliku prawym przyciskiem myszy, wybierz z menu podręcznego polecenie **Otwórz za pomocą** i wskaż odpowiedni edytor.

ja1.htm - Notepad2

Plik Edycja Widok Ustawienia Pomoc

```
<!DOCTYPE HTML>
<html>
   <head>
    <meta charset="utf-8">
    <meta name="description"content="Moja wizytówka">
    <meta name="keywords"content="Kleofas Słuszny, Kleo, Błądzigłowy, wizytówka, o mnie">
    <title>Cały ja</title>
   </head>
   <body>
                                      Cały ja
imię: Kleofas
nazwisko: Słuszny
miejsce zamieszkania: Błądzigłowy

Urodziłem się na początku XXI wieku. Wszyscy mówią, że to wiek szalony. Uważnie obserwuję
otaczający mnie świat i coraz częściej dochodzę do wniosku, że trudno jest zaprzeczyć tej
opinii.

Jaki będzie świat za 50 lat? Niestety tego nie wiem. Chciałbym jednak czynnie uczestniczyć
w jego kształtowaniu.
Myślę, że najważniejszą informacją o mnie będzie ta, że ponad wszystko cenię sobie spokój.

                                                              (jeszcze o mnie usłyszycie)
                                                                       Wasz Kleo
  </body>

</html>
```

Wiersz 24 : 24 Kolumna 8 Zaznaczono 0 1017 B UTF-8 CR+LF INS Strona internetowa

Rys. 3. Przykładowy dokument z podstawowymi znacznikami struktury

Aby tekst w przeglądarce wyświetlał się w postaci sformatowanej, należy zastosować odpowiednie znaczniki języka HTML, np.:

- **<h1>** i **</h1>** – początek i koniec nagłówka stopnia pierwszego (w języku HTML istnieje sześć stopni nagłówków; nagłówek stopnia pierwszego jest największy, a nagłówek stopnia szóstego – najmniejszy);
- **<p>** i **</p>** – odrębny akapit, oddzielony większym odstępem;
- **** i **** – tekst pogrubiony (znacznik ten odpowiada wyłącznie za wygląd danego wyrażenia; aby nadać mu specjalne znaczenie, dzięki któremu dany fragment treści będzie rozpoznawany przez sieciowe roboty, należy wykorzystać znacznik **<strong>**);
- **<i>** i **</i>** – tekst pochylony;
- **
** – nowy wiersz tekstu, ale nie odrębnego akapitu.

► Przeanalizuj poniższy zrzut, a następnie sformatuj odpowiednio swój dokument.

```
<!DOCTYPE HTML>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="description"content="Moja wizytówka">
    <meta name="keywords"content="Kleofas Słuszny, Kleo, Błądzigłowy, wizytówka, o mnie">
    <title>Cały ja</title>
  </head>
  <body>

    <h1>Cały ja</h1>
    <p>
      <b>imię:</b><i>Kleofas</i><br>
      <b>nazwisko:</b><i>Słuszny</i><br>
      <b>miejsce zamieszkania:</b><i>Błądzigłowy</i>
    </p>
    <p>
      Urodziłem się na początku XXI wieku. Wszyscy mówią, że to wiek szalony. Uważnie
obserwuję otaczający mnie świat i coraz częściej dochodzę do wniosku, że trudno jest
zaprzeczyć tej opinii.
    </p>
    <p>
      Jaki będzie świat za 50 lat? Niestety tego nie wiem. Chciałbym jednak czynnie
uczestniczyć w jego kształtowaniu. Myślę, że najważniejszą informacją o mnie będzie ta,
że ponad wszystko cenię sobie spokój.
    </p>
    <p>
      (jeszcze o mnie usłyszycie)<br>
      Wasz Kleo
    </p>

  </body>
</html>
```

Rys. 4. Przykładowy dokument z dodatkowymi znacznikami struktury

DEFINIOWANIE STYLÓW W DOKUMENCIE HTML

Za prezentację zawartości strony w przeglądarce odpowiadają Cascading Style Sheets (czytaj: kaskejding stajl szits) – kaskadowe arkusze stylów, w skrócie CSS. Pierwszą wersję tego języka opracowali w 1996 r. Håkon Wium Lie (czytaj: hokon wium li) i Bert Bos. Dzięki zastosowaniu CSS można m.in. sformatować czcionki i akapit, ustawić marginesy, wyrównania oraz dodać inne elementy ożywiające stronę.

Informacje na temat stylu, np. kroju czcionki, umieszcza się wewnątrz znaczników za pomocą atrybutu **`style=" "`**. Jeśli chcesz zastosować czcionkę Courier, należy napisać **`style="font-family: Courier;"`**. Kolejne właściwości i wartości wewnątrz cudzysłowu oddziela się średnikami. Jednostki zapisuje się bez odstępu, a nazwy składające się z kilku wyrazów wstawia się w apostrofy.

```
<p style="font-family: Courier;">
  W tym akapicie zastosowano czcionkę Courier.
</p>

<p style="font-family: 'Times New Roman';">
  W tym akapicie zastosowano czcionkę Times New Roman.
</p>
```

Rys. 5. Przykłady określenia kroju czcionki z wykorzystaniem atrybutu **`style`**

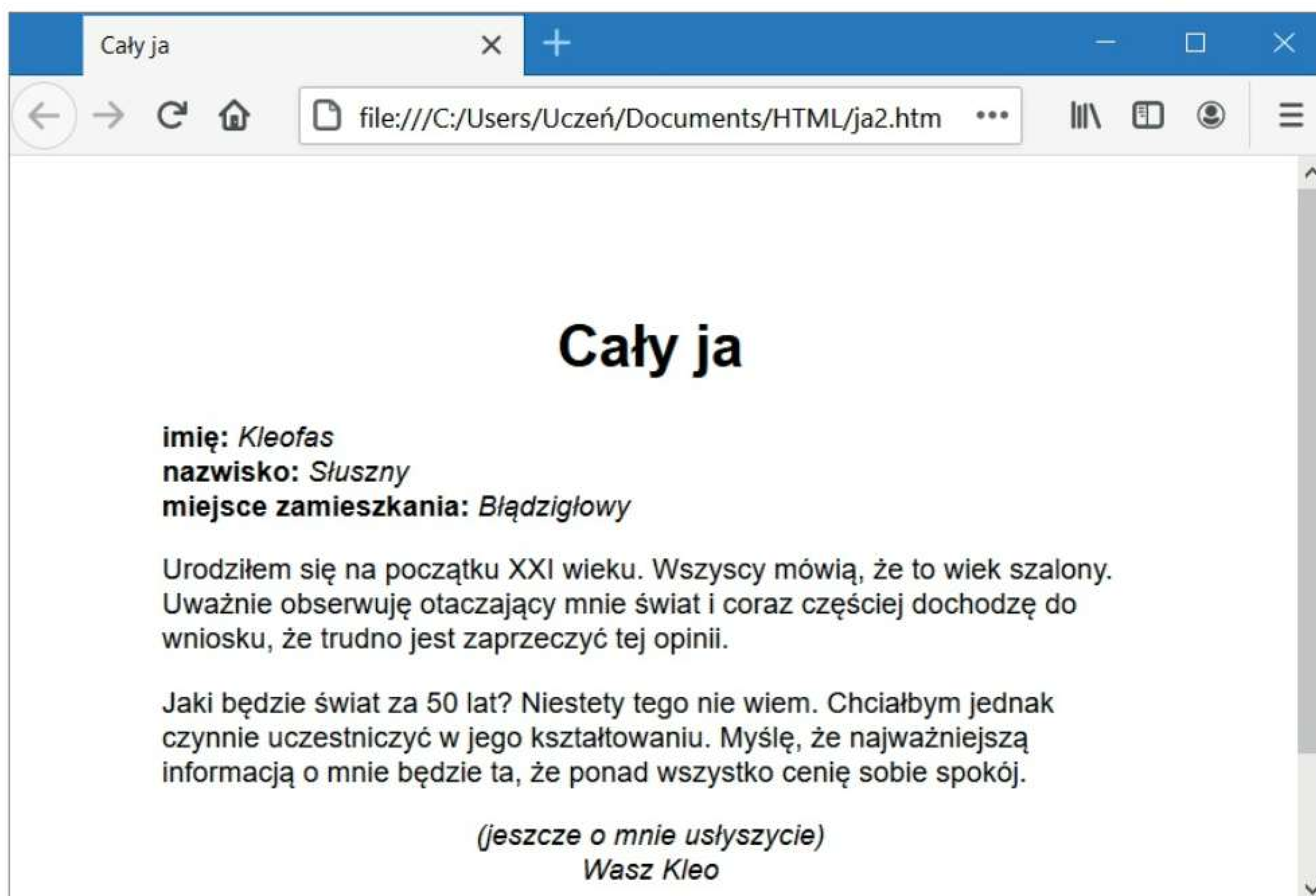
► Przeanalizuj zapisy dotyczące formatowania na poniższym zrzucie, a następnie w analogiczny sposób sformatuj swój dokument i zapisz go pod nazwą **ja2.htm**.

- Wiersz 9 – dla całego dokumentu określono krój czcionki (Arial), rozmiar czcionki (11 punktów) oraz szerokość marginesów (80 pikseli).
- Wiersz 11 – dla nagłówka stopnia pierwszego określono sposób wyrównania tekstu w poziomie (do środka).
- Wiersz 23 – dla ostatniego akapitu określono sposób wyrównania tekstu w poziomie (do środka) oraz styl czcionki (kursywa).

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="description"content="Moja wizytówka">
6     <meta name="keywords"content="Kleofas Słuszny, Kleo, Błądzigłowy, wizytówka, o mnie">
7     <title>Cały ja</title>
8   </head>
9   <body style="font-family: Arial; font-size: 11pt; margin: 80px;">
10
11    <h1 style="text-align: center;">Cały ja</h1>
12    <p>
13      <b>imię:</b><i>Kleofas</i><br>
14      <b>nazwisko:</b><i>Słuszny</i><br>
15      <b>miejsce zamieszkania:</b><i>Błądzigłowy</i>
16    </p>
17    <p>
18      Urodziłem się na początku XXI wieku. Wszyscy mówią, że to wiek szalony. Uważnie obserwuję otaczający mnie świat i coraz częściej dochodzę do wniosku, że trudno jest zaprzeczyć tej opinii.
19    </p>
20    <p>
21      Jaki będzie świat za 50 lat? Niestety tego nie wiem. Chciałbym jednak czynnie uczestniczyć w jego kształtowaniu. Myślę, że najważniejszą informacją o mnie będzie ta, że ponad wszystko cenię sobie spokój.
22    </p>
23    <p style="text-align: center; font-style: italic;">
24      (jeszcze o mnie usłyszycie)<br>
25      Wasz Kleo
26    </p>
27
28  </body>
29 </html>
```

Rys. 6. Przykładowy dokument z ostylowaniem

▶ Kliknij dwukrotnie plik **ja2.htm**, aby obejrzeć poprawioną wersję dokumentu w oknie przeglądarki.



Rys. 7. Przykładowy dokument – wersja ostateczna

ZADANIA

1. Otwórz kurs *Wstęp do HTML/CSS: Tworzenie stron internetowych* w Akademii Khana (**https://pl.khanacademy.org/computing/computer-programming/html-css**). Obejrzyj film *Witamy w sieci!*

2. Pozostań na stronie z zadania 1 i obejrzyj film *Podstawy HTML*. Po obejrzeniu filmu poeksperymentuj – wprowadź własny tekst w sekcji `body` i sformatuj go, używając rozwiązań poznanych w czasie lekcji.

3. Pozostań na stronie z zadania 1 i otwórz pierwsze wyzwanie. Napisz dwuzwrotkowy wiersz. Poprzedź go tytułem sformatowanym jako nagłówek (znacznik `h`). Zwrotki wiersza mają stanowić osobne akapity (znacznik `p`). Wewnątrz akapitów używaj przejść do następnej linii (znacznik `br`).

1.2 Prosta strona internetowa

DOWIESZ SIĘ, JAK
- definiować właściwości czcionek,
- stosować jednostki miary absolutne i względne,
- definiować właściwości akapitu.

Czas rozwinąć umiejętności dotyczące formatowania dokumentów HTML za pomocą kaskadowych arkuszy stylów. Nauczysz się nadawać określony wygląd czcionkom oraz akapitom i wykorzystasz najpopularniejsze narzędzia CSS, żeby stworzyć stronę poświęconą swojemu hobby.

KRÓJ CZCIONKI

Wartością właściwości **`font-family:`** może być dokładna nazwa czcionki (np. Arial, Calibri) lub jej ogólne określenie (**`serif`** – czcionka szeryfowa, **`sans-serif`** – czcionka bezszeryfowa, **`cursive`** – kursywa, **`monospace`** – czcionka o stałej szerokości, **`fantasy`** – czcionka ozdobna). Dobrze jest określić co najmniej dwa typy czcionek. Jeśli w komputerze użytkownika zabraknie typu występującego na pierwszej pozycji, przy wyświetlaniu zostanie zastosowany następny wskazany typ. Natomiast jeśli zabraknie wszystkich wymienionych czcionek, system wykorzysta domyślną czcionkę użytkownika. Poszczególne wartości wpisywane po nazwie właściwości oddziela się przecinkiem i spacją, a w przypadku nazw kilkuwyrazowych dodaje się apostrofy.

► Zdefiniuj styl w jednozdaniowym akapicie: preferowany krój czcionki – Courier New, w razie braku – czcionka o stałej szerokości albo bezszeryfowa.

```
<p style="font-family: 'courier new', monospace, 'sans-serif';">
  Ala ma kota.
</p>
```

```
Ala ma kota.
```

Rys. 1. Definicja kroju czcionki i jej rezultat

STYL CZCIONKI

Właściwość **`font-style:`** ma trzy dopuszczalne wartości:
- **`normal`** – czcionka prosta (jest to wartość domyślna);
- **`italic`** – kursywa dla zestawów czcionki ze znakami pochylonymi;
- **`oblique`** – kursywa dla zestawów czcionki bez znaków pochylonych.

▶ Zdefiniuj styl w dwóch jednozdaniowych akapitach: pierwszy akapit – zapis kursywą, drugi akapit – pochylona czcionka Courier New.

```
<p style="font-style: italic;">
  Ala ma słonia.
</p>
<p style="font-family: 'courier new'; font-style: italic;">
  Ala ma koguta.
</p>
```

Ala ma słonia.

Ala ma koguta.

Rys. 2. Definicje kroju i stylu czcionki oraz ich rezultat

WARIANT CZCIONKI

Właściwość **`font-variant:`** może przyjąć dwie wartości:

- **`normal`** – małe litery wyświetlające się normalnie (jest to wartość domyślna);
- **`small-caps`** – małe litery wyświetlające się w postaci kapitalików lub przeskalowanych wielkich liter.

▶ Zdefiniuj wariant czcionki w jednozdaniowym akapicie tak, aby małe litery wyświetlały się w postaci kapitalików.

```
<p style="font-variant: small-caps;">
  Ala ma kota, słonia i koguta.
</p>
```

ALA MA KOTA, SŁONIA I KOGUTA.

Rys. 3. Definicja wariantu czcionki i jej rezultat

WYSOKOŚĆ CZCIONKI

Właściwość **`font-size:`** dopuszcza poniższe wartości:

- **`medium`** – średni rozmiar czcionki (wielkość domyślna);
- **`small`** – wielkość będąca wynikiem działania **`medium / 1,5`**;
- **`x-small`** – wielkość będąca wynikiem działania **`small / 1,5`**;
- **`xx-small`** – wielkość będąca wynikiem działania **`x-small / 1,5`**;
- **`large`** – wielkość będąca wynikiem działania **`medium * 1,5`**;
- **`x-large`** – wielkość będąca wynikiem działania **`large * 1,5`**;
- **`xx-large`** – wielkość będąca wynikiem działania **`x-large * 1,5`**;
- **`larger`** – rozmiar czcionki większy niż ten przypisany do elementu nadrzędnego (wielkość względna);
- **`smaller`** – rozmiar czcionki mniejszy niż ten przypisany do elementu nadrzędnego (wielkość względna).

Jednostki absolutne i względne

Jednostki miary w dokumencie HTML stosuje się do określania nie tylko wielkości tekstu, lecz także szerokości i wysokości innych elementów, np. marginesów.

Jednostki absolutne

cm – centymetr
mm – milimetr
in – cal (1 in = 2,54 cm)
pt – punkt (1 pt = 1/72 in)
pc – pica (1 pc = 12 pt)
px – piksel

Jednostki względne

em – rozmiar liczony względem szerokości litery *m* w aktualnie używanej czcionce
ex – rozmiar liczony względem wysokości litery *x* w danej czcionce
% – rozmiar określony w procentach względem rozmiaru domyślnego

Jednostki absolutne odwołują się do konkretnej wartości numerycznej, są niezależne od rozdzielczości i wielkości ekranu. Jednostki względne odwołują się do rozmiaru innego elementu, co pozwala na lepsze dopasowanie w zależności od urządzeń wyjściowych. Domyślna wielkość czcionki w przeglądarkach to: 100% = 1 em = 16 px = = 12 pt = 1 pc.

► Zdefiniuj wysokość czcionki w jednozdaniowym akapicie tak, aby wysokość czcionki pierwszych dwóch wyrazów oraz kropki wynosiła 1 cm, a pozostały tekst został powiększony o 50%.

- Zastosuj znacznik **span**, który pozwala przypisać styl konkretnym elementom.

```
<p style="font-size: 1cm;">
  Ala ma <span style="font-size: 150%;">kota, słonia, koguta i jenota</span>.
</p>
```

Ala ma kota, słonia, koguta i jenota.

Rys. 4. Definicja wysokości czcionki i jej rezultat

ODSTĘPY MIĘDZY LITERAMI

Właściwość **letter-spacing:** może przyjąć dwie wartości: domyślną (**normal**) oraz liczbę w jednostkach miary. W przypadku wartości **normal** znaki tworzące dany tekst nie mają między sobą dodatkowego odstępu.

► Zdefiniuj odstępy między literami w trzech jednozdaniowych akapitach tak, aby odstępy w pierwszym akapicie wynosiły 2 mm, w drugim – 14 pikseli, a w trzecim były domyślne.

```
<p style="letter-spacing: 2mm;">
  Ala ma zwierzęta.
</p>
<p style="letter-spacing: 14px;">
  Ala ma zwierzęta.
</p>
<p style="letter-spacing: normal;">
  Ala ma zwierzęta.
</p>
```

```
A l a  m a  z w i e r z ę t a .

A  l  a   m  a   z  w  i  e  r  z  ę  t  a  .

Ala ma zwierzęta.
```

Rys. 5. Definicja odstępów między literami i jej rezultat

ZMIANA WIELKOŚCI ZNAKÓW

Jeśli chcesz zdefiniować zmianę wielkości znaków, musisz zastosować właściwość **`text-transform:`**, której dopuszczalne wartości to:

- **`capitalize`** – ustawienie wielkiej litery na początku każdego wyrazu;
- **`uppercase`** – wypisanie całego tekstu wielkimi literami;
- **`lowercase`** – wypisanie całego tekstu małymi literami;
- **`none`** – wartość domyślna.

► Zdefiniuj zmianę wielkości znaków w czterech jednozdaniowych akapitach tak, aby w pierwszym akapicie każdy wyraz rozpoczynał się wielką literą, w drugim cały tekst został wypisany wersalikami, w trzecim cały tekst został wypisany małymi literami, a w czwartym wielkość znaków miała wartość domyślną.

```
<p style="text-transform: capitalize;">
  Ala ma wiele różnych zwierząt.
</p>
<p style="text-transform: uppercase;">
  Ala ma wiele różnych zwierząt.
</p>
<p style="text-transform: lowercase;">
  Ala ma wiele różnych zwierząt.
</p>
<p style="text-transform: none;">
  Ala ma wiele różnych zwierząt.
</p>
```

```
Ala Ma Wiele Różnych Zwierząt.

ALA MA WIELE RÓŻNYCH ZWIERZĄT.

ala ma wiele różnych zwierząt.

Ala ma wiele różnych zwierząt.
```

Rys. 6. Definicja zmiany wielkości znaków i jej rezultat

ODSTĘPY MIĘDZY WYRAZAMI

Właściwość **`word-spacing:`** może przyjąć dwie wartości: domyślną (**`normal`**) oraz liczbę w jednostkach miary.

► Zdefiniuj odstępy między wyrazami w trzech jednozdaniowych akapitach tak, aby odstępy w pierwszym akapicie wynosiły 4 em, w drugim – 18 punktów, a w trzecim były domyślne.

```
<p style="word-spacing: 4em;">
  Ala ma zwierzaki.
</p>
<p style="word-spacing: 18pt;">
  Ala ma zwierzaki.
</p>
<p style="word-spacing: normal;">
  Ala ma  zwierzaki.
</p>
```

Ala   ma   zwierzaki.

Ala   ma   zwierzaki.

Ala ma zwierzaki.

Rys. 7. Definicja odstępów między wyrazami i jej rezultat

DEKOROWANIE TEKSTU

Jeśli chcesz podkreślić tekst, umieścić nad nim linię (nadkreślenie) lub go przekreślić, zastosuj właściwość **`text-decoration:`**, której dopuszczalne wartości to:

- **`underline`** – podkreślenie;
- **`overline`** – nadkreślenie;
- **`line-through`** – przekreślenie;
- **`none`** – brak elementów dekoracyjnych.

► Zdefiniuj dekorację w trzech jednozdaniowych akapitach tak, aby tekst w pierwszym akapicie był podkreślony, w drugim – nadkreślony, a w trzecim przekreślony.

```
<p style="text-decoration: underline;">
  Ala dba o swoje zwierzątka.
</p>
<p style="text-decoration: overline;">
  Ala dba o swoje zwierzątka.
</p>
<p style="text-decoration: line-through;">
  Ala dba o swoje zwierzątka.
</p>
```

Ala dba o swoje zwierzątka.

Ala dba o swoje zwierzątka.

~~Ala dba o swoje zwierzątka.~~

Rys. 8. Definicja dekoracji tekstu i jej rezultat

WYRÓWNYWANIE TEKSTU W POZIOMIE

Własność **`text-align:`** dopuszcza następujące wartości:

- **`left`** – wyrównanie tekstu do lewego marginesu elementu nadrzędnego;
- **`right`** – wyrównanie tekstu do prawego marginesu elementu nadrzędnego;
- **`center`** – wyrównanie tekstu względem środka elementu nadrzędnego;
- **`justify`** – wyrównanie tekstu jednocześnie do lewego i prawego marginesu elementu nadrzędnego.

► Zdefiniuj wyrównanie tekstu w poziomie do lewej, do prawej, do środka oraz do lewej i prawej. Określ szerokość akapitu lub bloku, tj. **`"width: "`**, na 235 pikseli. Sprawdź, co się stanie, jeśli justowanego tekstu będzie za dużo w danym akapicie lub bloku.

- Elementem nadrzędnym może być akapit **`<p>`** lub blok tekstu wydzielony za pomocą znaczników **`<div>`** lub **`<article>`**. Dwa ostatnie znaczniki tworzą bloki na wybranych fragmentach strony – z tym że znacznik **`<article>`** opisuje niezależny blok zawierający główną treść strony, natomiast znacznik **`<div>`** opisuje blok zawierający dowolną treść.

```
<p style="width: 235px; text-align: left;">
  Czy Ala chce zostać weterynarzem?<br>Nie, kapitanem żeglugi wielkiej.
</p>
<p style="width: 235px; text-align: right;">
  Czy Ala chce zostać weterynarzem?<br>Nie, dyrygentem.
</p>
<div style="width: 235px; text-align: center;">
  Czy Ala chce zostać weterynarzem?<br>Nie, archeologiem.
</div><br>
<div style="width: 235px; text-align: justify;">
  Czy Ala chce zostać weterynarzem?<br>Nie, programistą gier komputerowych.
</div>
```

Czy Ala chce zostać weterynarzem?
Nie, kapitanem żeglugi wielkiej.

Czy Ala chce zostać weterynarzem?
Nie, dyrygentem.

Czy Ala chce zostać weterynarzem?
Nie, archeologiem.

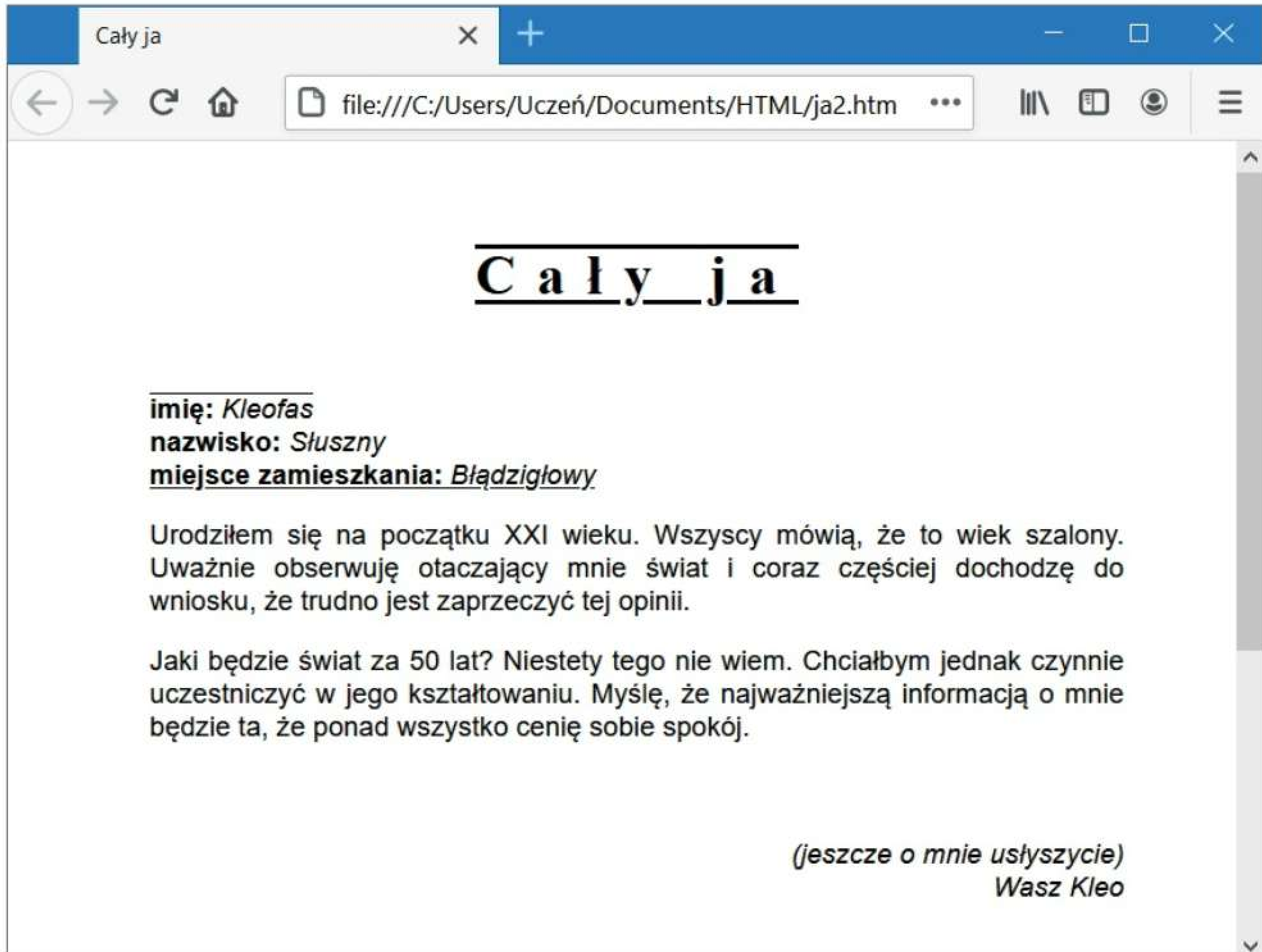
Czy Ala chce zostać weterynarzem?
Nie, programistą gier
komputerowych.

Rys. 9. Definicja wyrównania tekstu w poziomie i jej rezultat

► Utwórz dokument HTML o nazwie **mojehobby.htm**. Zadbaj o estetyczny wygląd dokumentu – wykorzystaj umiejętności zdobyte podczas tej i poprzedniej lekcji.

ZADANIE

1. Wykorzystując zdobytą wiedzę, udoskonal formatowanie dokumentu **ja2.htm** stworzonego podczas poprzedniej lekcji.



1.3 Strona w dobrym stylu

DOWIESZ SIĘ, JAK

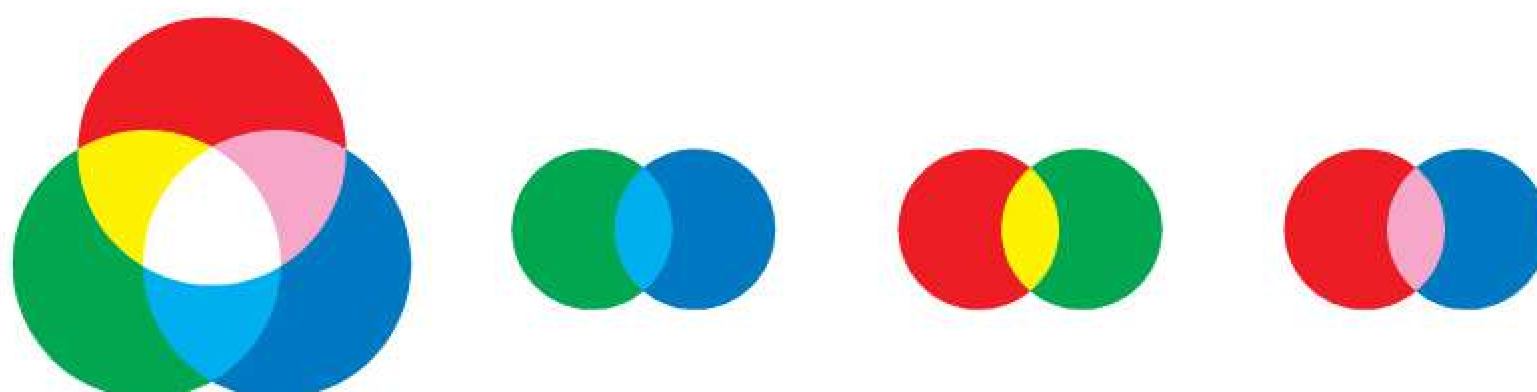
- definiować kolory tekstu, tła całej strony lub wybranego obszaru,
- osadzać elementy graficzne i umieszczać znaki specjalne,
- stosować wpisane, osadzone i zewnętrzne arkusze stylów.

To dalszy ciąg zajęć poświęconych kaskadowym arkuszom stylów. Dzięki wiadomościom zdobytym podczas tej lekcji nie będziesz mieć problemów z udoskonalaniem stron WWW.

DEFINIOWANIE KOLORÓW

Kolory w dokumencie HTML definiuje się na wiele sposobów. Można używać angielskich nazw kolorów albo stosować np. jeden z zapisów RGB – modelu przestrzeni barw opisywanej współrzędnymi określającymi udział czerwonego, zielonego i niebieskiego (**R**ed, **G**reen, **B**lue) w danym kolorze.

- Angielskie nazwy kolorów – na potrzeby języków HTML i CSS zdefiniowano nazwy 140 kolorów; używanie określeń spoza zdefiniowanej listy jest ryzykowne, bo choć przeglądarki rozpoznają obecnie wiele kolorów, nie ma gwarancji, że konkretne nazwy zostaną poprawnie zinterpretowane.
- Zapis szesnastkowy RGB nazywany heksadecymalnym – kod składa się ze znaku **#** oraz trzech dwuznakowych wartości odpowiadających za udział w danym kolorze barw czerwonej, zielonej i niebieskiej.
- Zapis dziesiętny RGB nazywany decymalnym – kod składa się z oznaczenia **`rgb`** oraz podanych w nawiasie współrzędnych informujących, jaka część podstawowej barwy zostanie użyta do otrzymania pożądanego koloru (wartości z zakresu 0–255).
- Zapis procentowy RGB – kod składa się z oznaczenia **`rgb`** oraz podanych w nawiasie wartości procentowych informujących, jaka część podstawowej barwy zostanie użyta do otrzymania pożądanego koloru (wartości z zakresu 0–100%).



Rys. 1. Model RGB

Poniższa tabela prezentuje różne formy zapisu nazw przykładowych kolorów.

Kolor	Nazwa angielska	Zapis szesnastkowy RGB	Zapis dziesiętny RGB	Zapis procentowy RGB
	black	#000000	rgb(0, 0, 0)	rgb(0%, 0%, 0%)
	white	#ffffff	rgb(255, 255, 255)	rgb(100%, 100%, 100%)
	red	#ff0000	rgb(255, 0, 0)	rgb(100%, 0%, 0%)
	yellow	#ffff00	rgb(255, 255, 0)	rgb(100%, 100%, 0%)
	fuchsia	#ff00ff	rgb(255, 0, 255)	rgb(100%, 0%, 100%)
	tomato	#ff6347	rgb(255, 99, 71)	rgb(100%, 38.3%, 27.8%)
	forestgreen	#228b22	rgb(34, 139, 34)	rgb(13.3%, 54.5%, 13.3%)

Tab. 1. Rożne formy zapisu nazw kolorów

Aby zdefiniować kolor tła strony lub jej fragmentu (np. nagłówka, bloku, paragrafu), należy odpowiednio opisać właściwość **background-color:**. Aby wskazać kolor czcionki w nagłówku, bloku czy akapicie, atrybut **style=" "** należy uzupełnić o właściwość **color:**. Aby opisać kolor czcionki pojedynczego wyrazu lub grupy wyrazów, w definicji należy dodatkowo uwzględnić znacznik **<span>**, który pozwala przypisać styl konkretnym elementom.

► Zdefiniuj tło w następującym kolorze: .

- Nazwa angielska: nie ma możliwości zapisu – brak zdefiniowanej nazwy.
- Zapis szesnastkowy RGB: **background-color: #aef188**.
- Zapis dziesiętny RGB: **background-color: rgb(174, 241, 136)**.
- Zapis procentowy RGB: **background-color: rgb(68.2%, 94.5%, 53.3%)**.

► Zdefiniuj kolorystykę tła i tekstu w jednozdaniowym akapicie zawierającym nazwy kolorów tak, aby wyświetlały się one w odpowiednich kolorach.

```
<p style="background-color: #ddddee; color: #000046;">
  Król Sobieski miał trzy pieski:
  <span style="color: #ff0000;">Czerwony</span>,
  <span style="color: #228b22;">Zielony</span>,
  <span style="color: #0000ff;">Niebieski</span>.
```

Król Sobieski miał trzy pieski: Czerwony, Zielony, Niebieski.

Rys. 2. Definicja kolorów tła i czcionki oraz jej rezultat

OSADZANIE ELEMENTÓW GRAFICZNYCH

W dokumencie HTML można umieścić elementy graficzne, m.in. ilustracje i obramowania różnych elementów.

Ilustracje osadza się za pomocą znacznika **`<img>`**. Aby wskazać plik, który ma zostać wyświetlony na stronie, należy posłużyć się atrybutem **`src`**.



Rys. 3. Podstawowy kod osadzający ilustrację w dokumencie – gdy obrazek znajduje się w tym samym folderze co dokument i gdy obrazek jest umieszczony w folderze obrazki

Opis alternatywny – wartość atrybutu alt

Tego typu opis jest wyświetlany, gdy obrazek nie może zostać wczytany przez przeglądarkę. Jeśli użytkownik korzysta ze specjalnych programów dla osób niewidomych lub słabowidzących, opis ten jest odczytywany jako normalny tekst na stronie.

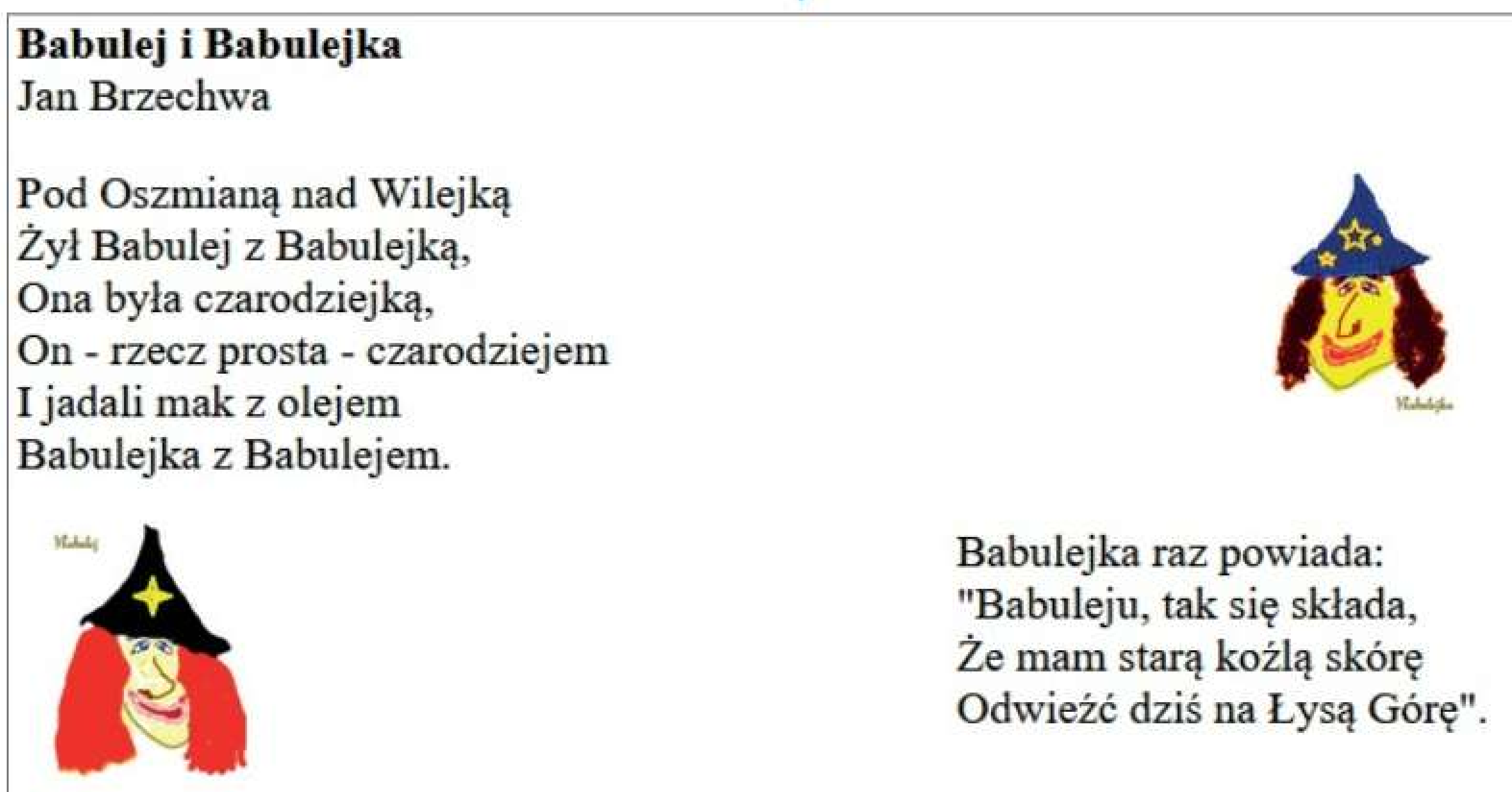
Podstawowy zapis można rozszerzyć o atrybuty stylu:

- **`float:`** – określa sposób wyrównywania obrazu: **`right`** – do prawej krawędzi obszaru, **`left`** – do lewej (takie rozwiązanie pozwala na oblewanie obrazu tekstem), **`none`** – obraz będzie traktowany jak jeden ze znaków w tekście (jest to wartość domyślna);
- **`height:`** – określa wysokość obrazu (szerokość dostosuje się proporcjonalnie);
- **`margin-left:`** lub **`margin-right:`** – określa odległość od lewego lub prawego marginesu obrazu;
- **`margin-top:`** lub **`margin-bottom:`** – określa odległość od górnego lub dolnego marginesu obrazu.

► Przeanalizuj kod i zaproponuj, jak go uzupełnić, aby osiągnąć efekt zaprezentowany na zrzucie poniżej.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
      <title>Babulej i Babulejka</title>
  </head>
  <body>
    <article style="width: 520px; height: 320px;">
      <p><b>Babulej i Babulejka</b><br>Jan Brzechwa</p>
      <img src="ona.jpg" style="height: 90px; float:..........; margin-right: 10px;"
       alt="Babulejka">
      <p>
        Pod Oszmianą nad Wilejką<br>
        Żył Babulej z Babulejką,<br>
        Ona była czarodziejką,<br>
        On - rzecz prosta - czarodziejem<br>
        I jadali mak z olejem<br>
        Babulejka z Babulejem.
      </p>
      <img src="on.jpg" style="height: 90px; float:.........; margin-left: 10px;
       margin-right: 250px" .....="Babulej">
      <p>
        Babulejka raz powiada:<br>
        "Babuleju, tak się składa,<br>
        Że mam starą koźlą skórę<br>
        Odwieźć dziś na Łysą Górę".
      </p>
    </article>
  </body>
</html>
```





Rys. 4. Kod z lukami i oczekiwany rezultat

WPROWADZANIE OBRAMOWAŃ ELEMENTÓW

Aby wprowadzić obramowanie danego obszaru, należy ustalić wartości właściwości **`border:`**, tj. grubość (np. w pikselach), styl (np. **`dotted`** – linia kropkowana, **`dashed`** – linia kreskowana, **`solid`** – linia ciągła, **`double`** – linia podwójna) i kolor. Czasem

konieczne jest również ustawienie odstępu między krawędzią elementu a jego zawartością (**`padding:`**).

▶ Przeanalizuj kod. Co zostało zmienione w kodzie źródłowym i jaki uzyskano efekt?

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
      <title>Babulej i Babulejka</title>
  </head>
  <body>
    <article style="width: 500px; height: 300px; border: 4px double #008d00;">
      <div style="background-color: #008d00; padding: 10px; margin: 0;">
        <b>Babulej i Babulejka</b><br>Jan Brzechwa
      </div>
      <img src="ona.jpg" style="height: 100px; float: right; margin-right: 10px;"
       alt="Babulejka">
      <p>
        Pod Oszmianą nad Wilejką<br>Żył Babulej z Babulejką,<br>
        Ona była czarodziejką,<br>On - rzecz prosta - czarodziejem<br>
        I jadali mak z olejem<br>Babulejka z Babulejem.
      </p>
      <img src="on.jpg" style="height: 100px; float: left;  margin-left: 10px;
       margin-right: 200px" alt="Babulej">
      <p>
        Babulejka raz powiada:<br>"Babuleju, tak się składa,<br>
        Że mam starą koźlą skórę<br>Odwieźć dziś na Łysą Górę".
      </p>
    </article>
  </body>
</html>
```

Rys. 5. Fragment kodu strony z obramowaniem

ZNAKI SPECJALNE

Aby do dokumentu wstawić znak niedostępny na klawiaturze, trzeba się posłużyć kodami poszczególnych znaków. Każdy znak specjalny należy poprzedzić znakiem **&** i zakończyć średnikiem. Istnieją trzy sposoby wprowadzania znaków specjalnych: za pomocą ich nazw, zapisu dziesiętnego lub zapisu szesnastkowego.

Jak uzyskać znak €?		
zapis z użyciem nazwy znaku	zapis dziesiętny	zapis szesnastkowy
€	€	€

Tab. 2. Różne sposoby zapisu symbolu euro

▶ Znajdź w internecie zapis dziesiętny lub szesnastkowy cudzysłowu w formacie „”. Wypróbuj go.

Na szczególną uwagę zasługuje znak **` `**. Pozwala on uniknąć pozostawienia wiszących liter na końcu wierszy. Działa tak samo jak twarda spacja w edytorze tekstu.

► Przeanalizuj poniższy przykład.

```
<p>
  Tak dalece, jak to możliwe, nie wyrzekając się
  siebie, bądź w dobrych  stosunkach z innymi ludźmi.
  Prawdę swą głoś spokojnie i jasno, słuchaj też
  tego, co mówią inni, nawet głupcy i ignoranci, oni
  też mają swoją opowieść. Jeśli porównujesz się
  z innymi, możesz stać się próżny lub zgorzkniały,
  albowiem zawsze będą lepsi i gorsi od Ciebie.
</p>
```

Tak dalece, jak to możliwe, nie wyrzekając się siebie, bądź w
dobrych stosunkach z innymi ludźmi. Prawdę swą głoś spokojnie
i jasno, słuchaj też tego, co mówią inni, nawet głupcy i
ignoranci, oni też mają swoją opowieść. Jeśli porównujesz się z
innymi, możesz stać się próżny lub zgorzkniały, albowiem
zawsze będą lepsi i gorsi od Ciebie.

```
<p>
  Tak dalece, jak to możliwe, nie wyrzekając się
  siebie, bądź w dobrych stosunkach
  z innymi ludźmi. Prawdę swą głoś spokojnie
  i jasno, słuchaj też tego, co mówią inni, nawet
  głupcy i ignoranci, oni też mają swoją opowieść.
  Jeśli porównujesz się z innymi, możesz stać się
  próżny lub zgorzkniały, albowiem zawsze będą lepsi
  i gorsi od Ciebie.
</p>
```

Tak dalece, jak to możliwe, nie wyrzekając się siebie, bądź
w dobrych stosunkach z innymi ludźmi. Prawdę swą głoś
spokojnie i jasno, słuchaj też tego, co mówią inni, nawet głupcy
i ignoranci, oni też mają swoją opowieść. Jeśli porównujesz się
z innymi, możesz stać się próżny lub zgorzkniały, albowiem
zawsze będą lepsi i gorsi od Ciebie.

Rys. 6. Kod źródłowy bez znaku twardej spacji i ze znakiem twardej spacji oraz wyświetlający się tekst

RODZAJE KASKADOWYCH ARKUSZY STYLÓW

Style wpisane, stosowane do tej pory podczas lekcji, określają sposób wyświetlania tylko tego elementu, którego dotyczy dany znacznik, i nie wpływają na sposób wyświetlania innych elementów, nawet tego samego typu (można się o tym przekonać, analizując kod na rys. 5). W przypadku bardziej złożonych projektów jest to metoda zbyt pracochłonna, poza tym kod szybko staje się obszerny i nieczytelny.

Style osadzone umieszcza się w części nagłówkowej dokumentu. W tym przypadku wystarczy raz zdefiniować styl określonego elementu, a w całym dokumencie będzie on obowiązywał we wszystkich elementach tego samego typu. Jeśli witryna składa się z kilku dokumentów, to w każdym z nich trzeba powtarzać definicję stylu.

```
<style>
  article {
  border: 4px double #003800;
  padding: 0;
  width: 514px;
  }
  p.tytuł {
  background-color: #008d00;
  color: #ffffe9;
  padding: 8px;
  width: 500px;
  font-size: 16pt;
  font-family: 'Times New Roman';
  }
  p {
  width: 500px;
  font-size: 12pt;
  font-family: arial;
  color: #000094;
  padding: 10px;
  margin: 0;
  }
</style>
```

sekcja **head**

```
<article>
  <p class="tytuł">
    <b>Babulej i Babulejka</b><br>Jan Brzechwa
  </p>
  <p>
    Pod Oszmianą nad Wilejką<br>
    Żył Babulej z Babulejką,<br>
    Ona była czarodziejką,<br>
    On - rzecz prosta - czarodziejem<br>
    I jadali mak z olejem<br>
    Babulejka z Babulejem.
  </p>
  <p>
    Babulejka raz powiada:<br>
    "Babuleju, tak się składa,<br>
    Że mam starą koźlą skórę<br>
    Odwieźć dziś na Łysą Górę".
  </p>
</article>
```

sekcja **body**

Rys. 7. Przykład stylu osadzonego

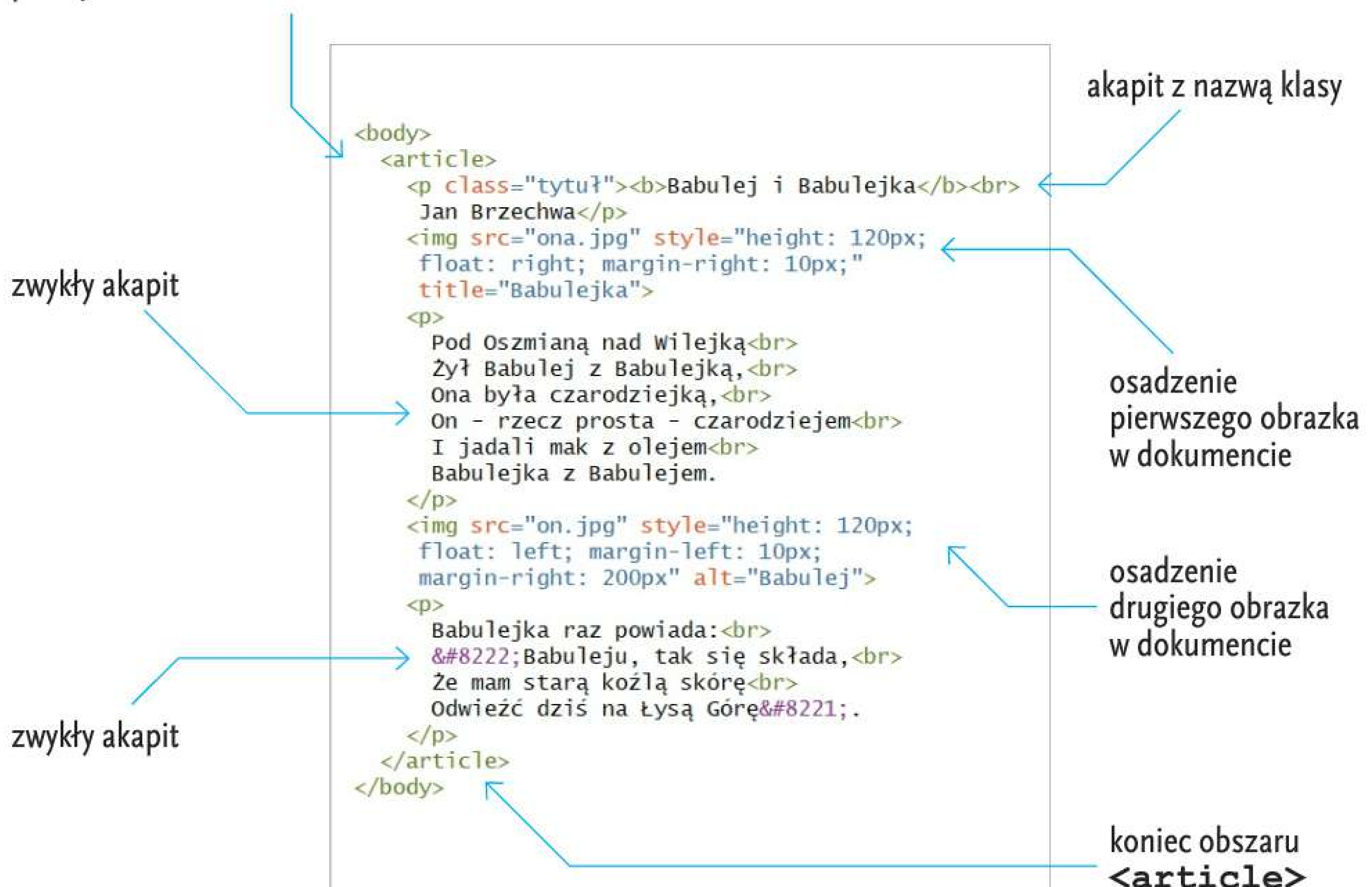
Arkusz stylów zewnętrznych to plik tekstowy z rozszerzeniem .css (np. **mojstyl.css**), w którym są zapisane definicje stylów. Oznacza to, że wystarczy zdefiniować style w jednym pliku i odwoływać się do niego we wszystkich dokumentach witryny, aby dokumenty te uzyskały określony, jednakowy wygląd. W części **`head`** dokumentu HTML należy wpisać adres pliku stylu. Do tego celu wykorzystuje się znacznik **`<link>`**.

```
<link rel="stylesheet" href="mojstyl.css" type="text/css">
```

Rys. 8. Przykład odnośnika do pliku stylu

W jaki sposób wprowadzić do dokumentu styl osadzony? Trzeba dobrze przemyśleć strukturę dokumentu – określić bloki typu **`<article>`** i **`<div>`** oraz akapity, do których należy się odwoływać w definicjach stylu w części **`head`**. Konieczne może okazać się stworzenie bloków lub akapitów z klasą (np. **`.tytuł`**), tak by oznaczony w ten sposób blok lub akapit był automatycznie ostylowany w opisany sposób.

> **Czym jest klasa?**
>
> Dzięki klasom można nadać grupie pewnych elementów HTML (np. tytułom) konkretny styl. Selektory klasowe tworzy się przez poprzedzenie nazwy klasy kropką (np. **`.tytuł`**).



Rys. 9. Przykład sekcji **`body`**

Po wpisaniu treści do sekcji **body** można odpowiednio uzupełnić sekcję nagłówkową.

początek deklaracji stylu osadzonego

```
<style>

  article {
  border: 4px double #003800;
  padding: 0;
  width: 500px;
  }

  p.tytuł {
  width: 500px;
  background-color: #008d00;
  font-family: 'Times New Roman';
  font-size: 16pt;
  color: #ffffe9;
  padding: 8px;
  }

  p {
  width: 500px;
  font-family: Arial;
  font-size: 12pt;
  color: #000094;
  padding: 10px;
  margin: 0;
  }

</style>
```

definicja obramowania dla obszaru **<article>**

Definicja stylu akapitu z klasą **.tytuł**. Każdy akapit, któremu nadano klasę **.tytuł**, będzie miał szerokość 500 px i tło w kolorze #008d00. Tekst zostanie wypisany czcionką Times New Roman o wysokości 16 pt i w kolorze #ffffe9. Odstęp od krawędzi do tekstu nie będzie mniejszy niż 8 px.

Definicja stylu dla zwykłego akapitu. Każdy zwykły akapit będzie miał szerokość 500 px. Tekst zostanie wypisany czcionką Arial wysokości 12 pt i w kolorze #000094. Odstęp od krawędzi akapitu do tekstu nie będzie mniejszy niż 10 px.

koniec deklaracji stylu osadzonego

Rys. 10. Przykład uzupełnionej części nagłówkowej

ZADANIA

1. Zmodyfikuj dokumenty **ja2.htm** i **mojehobby.htm** – zamień style wpisane na osadzone, zilustruj tekst stosownymi elementami graficznymi oraz popraw estetykę dokumentu.

2. Otwórz kurs *Wstęp do HTML/CSS: Tworzenie stron internetowych* w Akademii Khana, znajdź temat *Wprowadzenie do HTML* i zadanie zatytułowane *Wyzwanie: Doskonała wycieczka w obrazkach*. W razie trudności z rozwiązaniem zadania zapoznaj się z poprzedzającą je teorią.

3. Otwórz kurs *Wstęp do HTML/CSS: Tworzenie stron internetowych* w Akademii Khana, znajdź temat *Wprowadzenie do CSS* i zadanie zatytułowane *Wyzwanie: Klasy jabłek i bananów*. W razie trudności z rozwiązaniem zadania zapoznaj się z poprzedzającą je teorią.

1.4 Strona interaktywna

DOWIESZ SIĘ, JAK
- tworzyć elementy interaktywne,
- zbudować galerię zdjęć z wykorzystaniem elementów interaktywnych.

Dawniej strona ładowała się, wyświetlała i koniec – pozostawała statyczna. Podczas tej lekcji poznasz kilka rozwiązań, które umożliwiają wzajemne oddziaływanie człowieka i komputera.

INTERAKCJE Z WYKORZYSTANIEM CSS
Kaskadowe arkusze stylów oferują wiele możliwości wpływania na wygląd tekstu. W celu uzyskania interakcji najczęściej stosuje się tzw. **pseudoklasy**, które pozwalają zdefiniować w arkuszu stylów dynamiczne zdarzenia dotyczące konkretnych elementów w dokumencie HTML. Pseudoklasa `:hover` powoduje zmianę wskazanych w kodzie parametrów elementu po najechaniu na ten element kursorem myszy.

Aby podświetlić tekst, czyli zmienić kolor tła akapitu, do deklaracji stylu należy dodać pseudoklasę `p:hover` i odpowiednio opisać kolory tła oraz tekstu.



Rys. 1. Deklaracja zmiany koloru tła na czarny i koloru czcionki na biały

► Jak powinien wyglądać zapis w sekcji **body** dokumentu i w deklaracji stylu, aby po najechaniu kursorem myszy na tekst osiągnąć poniższy efekt?



Rys. 2. Efekt podświetlenia tekstu

W podobny sposób można uzyskać podświetlenie przycisku, tj. zmianę koloru przycisku oraz umieszczonego na nim tekstu.

Rys. 3. Przykładowy efekt podświetlenia przycisku po najechaniu na niego kursorem

► Przeanalizuj poniższy zapis. Zauważ, że w deklaracji stylu określono typ kursora – **`pointer`**, tj. wskaźnik. Właściwość **`cursor:`** może przyjąć też postać celownika (**`crosshair`**) czy lupki z plusikiem (**`zoom-in`**), wskazującej np. na możliwość powiększenia zdjęcia.

Zapis w deklaracji stylu

```
div.przycisk {
  background-color: #C6C3C4;
  color: black;
  width: 220px;
  height: 34px;
  padding: 6px;
  font-size: 34px;
  margin: 3px;
  text-align: center;
 }
div.przycisk:hover {
  background-color: #918C8C;
  color: red;
  cursor: pointer;
}
```

Zapis w sekcji body dokumentu

```
<div class="przycisk">
   A l a  D u ż a
</div>

<div class="przycisk">
   O l a  M a ł a
</div>

<div class="przycisk">
   J a n  Ś r e d n i
</div>
```

Rys. 4. Wykorzystanie pseudoklasy **`:hover`** w definicji stylu w celu stworzenia interaktywnych elementów

► Jakie zmiany należałoby wprowadzić w powyższej deklaracji stylu, aby po najechaniu kursorem myszy na przycisk tekst zmniejszył się o 14 pt?

Ciekawą interaktywną opcją jest też zmiana grafiki w tle.

Rys. 5. Przykładowy efekt zmiany grafiki w tle

Blok z fotografią **<div class="fotka"></div>** umieszczony w sekcji **body** można opisać w deklaracji stylu tak, jak pokazano poniżej.

Zapis w deklaracji stylu

```
div.fotka {
   background-image: url('foka1.jpg');
   background-repeat: no-repeat;
   background-position: center;
   width: 400px;
   height: 300px;
   display: block;
   border: 1px solid black;
   position: absolute;
   top: 40px;
   left: 40px;
  }

div.fotka:hover {
   cursor: pointer;
   background: url('foka2.jpg');
   background-repeat: no-repeat;
   background-position: center;
  }
```

Zapis w sekcji body dokumentu

```
<div class="fotka"></div>
```

Rys. 6. Deklaracja zmiany grafiki w tle po najechaniu kursorem myszy **<div class="fotka"></div>**

Za pomocą właściwości **background-image:** definiuje się tło graficzne danego obszaru. Dopuszczalne są następujące wartości:

- **url** – określa adres pliku graficznego mającego stanowić tło obszaru;
- **none** – określa, że dany element nie powinien mieć obrazka w tle;
- **inherit** – określa, że dany element ma dziedziczyć tło elementu nadrzędnego.

Czym jest URL?

Skrót URL najczęściej jest kojarzony z adresami stron WWW (np. **http://www.wsip.pl/wsipnet-pl/**), ale jest to format określania lokalizacji wszelkiego typu zasobów dostępnych w internecie. Aby wskazać adres obrazka, który ma zostać wyświetlony na stronie, należy użyć słowa kluczowego url i wpisać adres, np. `url(https://zkomputerem.pl/face/HTML/graf/zdjecia/Zoo/1.jpg)`.

Właściwość **background-repeat:** określa sposób powtarzania się obrazka tworzącego tło. Dopuszczalne są następujące wartości:

- **repeat** – obrazek powtarzający się w poziomie i w pionie;
- **repeat-x** – obrazek powtarzający się tylko w poziomie;
- **repeat-y** – obrazek powtarzający się tylko w pionie;
- **no-repeat** – obrazek niepowtarzający się.

Właściwość **background-position:** służy do opisania położenia obrazka względem elementu, którego ten obrazek jest tłem. Można zadać odpowiednią wartość, np. **center** wyśrodkowuje w pionie i poziomie obrazek stanowiący tło elementu, albo wskazać odległości lewego górnego narożnika obrazka od lewej i górnej krawędzi elementu, tak jak zaprezentowano na rys. 7.



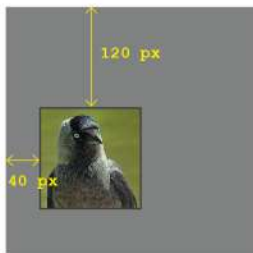
Zapis w deklaracji stylu

```
div.foto {
  background-image: url(m8.jpg);
  background-repeat: no-repeat;
  background-position: 40px 120px;
  background-color: gray;
  width: 300px;
  height: 300px;
 }
```

Zapis w sekcji body dokumentu

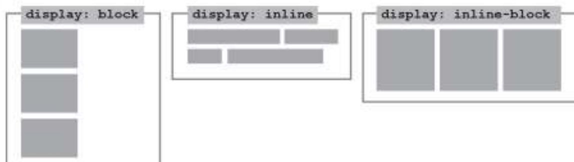
```
<div class="foto"></div>
```

Efekt uzyskany w oknie przeglądarki



Rys. 7. Pozycjonowanie niepowtarzającego się tła graficznego względem lewej i górnej krawędzi obszaru

Za pomocą właściwości **`display:`** można określić m.in. sposób układania się danego elementu HTML względem sąsiadujących elementów. Wartość **`inline`** sprawia, że elementy układają się w poziomie jeden obok drugiego; wartość **`block`** pozwala umieścić elementy w pionie jeden pod drugim względem pozostałych elementów.



Rys. 8. Przykładowe układy treści na stronie

Za pomocą właściwości **`border: 1px solid black;`** skraca się długość definicji stylu. Ten zapis ustala obramowanie o grubości 1 px linią ciągłą w kolorze czarnym.



Rys. 9. Porównanie dwóch sposobów definiowania obramowania elementu

Do odpowiedniego zdefiniowania efektu zmiany grafiki w tle konieczne może okazać się również ustalenie na stronie pozycji elementu blokowego, tj. elementu określonego znacznikami **`<article>`**, **`<div>`**, **`<p>`**. Służą do tego m.in. właściwości stylów, takie jak: **`position:`**, **`top:`** oraz **`left:`**. Właściwość **`position:`** najczęściej przyjmuje wartości **`absolute`** lub **`relative`**. Właściwości takie jak **`top:`** czy **`left:`** najczęściej przyjmują wartości określane w pikselach – **`px`**. Właściwość **`position: absolute;`** określa położenie elementu względem elementu nadrzędnego. Właściwość

position: relative określa położenie elementu względem jego pozycji spoczynkowej (takiej, jaką miałby bez wprowadzania zmian jego położenia).

```
<body>
  <div style="display: block; width: 200px;
  height: 200px; background-color: red;">
    <div style="display: block; width: 100px;
     height: 100px; position: absolute;
     top: 50px; left: 50px;
     background: yellow;"></div>
  </div>
</body>
```





Rys. 10. Przykład pozycjonowania obszaru **div** względem obszaru nadrzędnego

► Zastanów się, jakie zmiany należy wprowadzić w kodzie przedstawionym na rys. 10, aby otrzymać taki efekt jak na rys. 11.

```
<body>
  <div style="display: block; width: 200px;
  height: 200px; background-color: red;">
    <div style="display: block; width: 100px;
     height: 100px; ??????????????????????
     ???????????????????????????????????
     background: yellow;"></div>
  </div>
</body>
```





Rys. 11. Zmiana pozycjonowania obszaru z rys. 10

Przydatnym efektem bywa też zmiana przezroczystości obrazu i wyświetlenie etykietki z opisem. Stopień przezroczystości można określić za pomocą właściwości **opacity** przez nadanie jej wartości liczbowej z zakresu od **0** do **1**, przy czym wartość **0** wyłącza przezroczystość obiektu, a wartość **1** zapewnia całkowitą przezroczystość. Informację tekstową można dodać za pomocą atrybutu **title** – tekst wyświetla się w wyskakującym okienku po najechaniu kursorem myszy na ten obiekt.

Czym się różni title od alt?

W przeciwieństwie do **alt** atrybut **title** nie wspomaga robotów wyszukiwarek internetowych w odszukiwaniu obrazów. Choć wiele osób po prostu powiela treść w jednym i drugim miejscu, zgodnie z zamysłem atrybut **title** powinien udostępniać jedynie informacje dodatkowe.



Rys. 12. Przykład zmiany przezroczystości i wyświetlenia etykietki po najechaniu kursorem myszy na obrazek

▶ Przeanalizuj poniższy zapis.

Zapis w deklaracji stylu

```
img.fot:hover {
 opacity: 0.5;
}
```

Zapis w sekcji body dokumentu

```
<img class="fot" src="sokolica.jpg" width="300px"
 title="Przezroczystość: 50 procent">
```

Rys. 13. Zapis powodujący zmianę wyglądu zdjęcia po najechaniu kursorem myszy

▶ Przeanalizuj definicję stylu przedstawioną na rys. 14. Jak twoim zdaniem będzie wyglądał obrazek, zanim najedziesz na niego kursorem myszy, a jak po tym, gdy wykonasz tę czynność?

```
img {
 opacity: 0.4;
 }

img:hover {
 opacity: 1.0;
 }
```

Rys. 14. Przykład ostylowania obrazka – zmiana wyglądu następuje po najechaniu kursorem myszy

INTERAKCJA Z WYKORZYSTANIEM ZDARZEŃ JAVASCRIPT

Projektanci stron WWW wykorzystują w swojej pracy nie tylko HTML i CSS, lecz także języki skryptowe, np. PHP czy JavaScript, pozwalające na umieszczenie na stronie bardziej zaawansowanych interakcji z użytkownikiem, animacji czy formularzy.

Język JavaScript umożliwia wykorzystywanie w dokumencie HTML zdarzeń:
- **onmouseover** – gdy kursor myszy ustawiono na danym elemencie;
- **onmouseout** – gdy kursor myszy przesunięto poza obszar danego elementu;
- **onclick** – gdy kliknięto lewy przycisk myszy.

Poniżej zaprezentowano, w jaki sposób zaprogramować zmianę wyglądu przycisku w reakcji na ustawienie na nim kursora myszy i przesunięcie kursora poza przycisk.

► Przeanalizuj zapis w sekcji **body** dokumentu.

Zapis w sekcji body dokumentu

```
<img src="p1.jpg" onmouseover="this.src='p2.jpg'" onmouseout="this.src ='p1.jpg'">
```

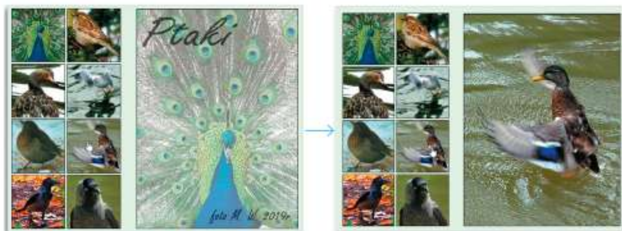
Efekt

bez interakcji	po najechaniu kursorem myszy na obraz	po przesunięciu kursora poza obszar obrazu
		

Rys. 15. Zastosowanie zdarzeń języka JavaScript do tworzenia przycisków animowanych

► Jakie zmiany należałoby wprowadzić, aby po kliknięciu lewym przyciskiem myszy na przycisku pojawił się napis „Dalej!"?

Dzięki zastosowaniu zdarzeń języka JavaScript można również stworzyć galerię. W poniższym przykładzie przyjęto założenie, że obrazki (**obraz1.jpg**, **obraz2.jpg**, ... **obraz8.jpg**) będą pokazywane po kliknięciu odpowiedniej miniatury (**m1.jpg**, **m2.jpg**, ... **m8.jpg**), przygotowanej wcześniej w programie graficznym.



Rys. 16. Przykładowy pokaz obrazków: zdjęcie tytułowe i zdjęcie wyświetlane po kliknięciu miniaturki



▶ Przeanalizuj zapis wprowadzający zmianę obrazu w reakcji na zdarzenie **onclick**.

Zapis w deklaracji stylu

```
body {
    margin: 0;
    background-color: #E9FFE9;
  }
div {
    display: block;
    position: relative;
  }
.rama {
    width: 632px;
    margin: 5% auto auto auto;
    padding: 0;
  }
.mini {
    width: 246px;
    left: 0;
    top: 0;
    padding: 0;
    float: left;
  }
.m {
    cursor: pointer;
  }
.obraz {
    background-image: url('obraz0.jpg');
    background-repeat: no-repeat;
    background-position: center;
    height: 492px;
    width: 360px;
    left: 278px;
  }
```

Zapis w sekcji body dokumentu

```
<div class="rama">

  <div class="mini">

    <img class="m" src="m1.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz1.jpg)';">
    <img class="m" src="m2.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz2.jpg)';">
    <br>

    <img class="m" src="m3.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz3.jpg)';">
    <img class="m" src="m4.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz4.jpg)';">
    <br>

    <img class="m" src="m5.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz5.jpg)';">
    <img class="m" src="m6.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz6.jpg)';">
    <br>

    <img class="m" src="m7.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz7.jpg)';">
    <img class="m" src="m8.jpg"
     onclick="document.getElementById('obraz').style.backgroundImage='url(obraz8.jpg)';">
  </div>

  <div class="obraz" id="obraz">
     
  </div>

</div>
```

Rys. 17. Ostylowanie galerii zdjęć i zawartość sekcji **body**

ZADANIA

1. Przeanalizuj kod źródłowy wskazanych poniżej trzech galerii zdjęć pod kątem zastosowanych rozwiązań:

- **https://zkomputerem.pl/face/HTML/8.1.3/interjs/pokaz1/pokaz1.htm**;
- **https://zkomputerem.pl/face/HTML/8.1.3/zooimg/galeria.htm**;
- **https://zkomputerem.pl/face/HTML/8.1.3/zootlo/galeria3.htm**.

Aby odczytać kod źródłowy strony, kliknij ją prawym przyciskiem myszy i z menu podręcznego wybierz polecenie **Pokaż źródło strony**, **Wyświetl źródło strony** lub podobnie brzmiące polecenie (tekst tego polecenia zależy od przeglądarki, której używasz).



2. Utwórz dokument **galeria.htm**. Zaprezentuj w nim przynajmniej sześć zdjęć. Zadbaj o funkcjonalność i estetykę wykonanej pracy.

3. Otwórz kurs *Wstęp do HTML/CSS: Tworzenie stron internetowych* w Akademii Khana i znajdź temat *Więcej selektorów CSS*. Zapoznaj się z informacjami dotyczącymi dynamicznych pseudoklas w CSS, a następnie wykonaj zadanie zatytułowane *Wyzwanie: Zgrupowane zwierzęta*.

1.5 Witryna WWW

DOWIESZ SIĘ,
- jakie są rodzaje witryn,
- jak uporządkować kod dokumentów tworzących witrynę,
- jak połączyć dokumenty składowe witryny systemem odnośników.

W internecie funkcjonują zarówno pojedyncze strony, jak i złożone witryny. Podczas tej lekcji połączysz przygotowane do tej pory dokumenty HTML odnośnikami i stworzysz witrynę złożoną z trzech stron.

CZYM JEST WITRYNA WWW

Witryna WWW, zwana również serwisem WWW, to zbiór powiązanych ze sobą stron WWW i innych dokumentów (np. plików PDF) umieszczonych na jednym serwerze w danej domenie, tworzących spójną pod względem szaty graficznej, rozbudowaną publikację. Potocznie witrynę WWW często nazywa się stroną internetową, a prostą, pojedynczą stronę tej witryny – podstroną.

Domena

Adres WWW serwisu składa się z nazwy głównej (np. nazwy firmy) oraz rozszerzenia, czyli domeny:
- podstawowej, np. .com, .net, .org, .gov, .edu, .info;
- krajowej, np. .pl (Polska), .de (Niemcy), .uk (Wielka Brytania), .eu (Unia Europejska), .ru (Rosja), .fr (Francja);
- regionalnej, np. .warszawa.pl, .waw.pl, .czest.pl, .slask.pl;
- branżowej, np. .hotel, .auto, .coffee, .club;
- funkcjonalnej narodowej, np. .com.pl, .nieruchomosci.pl, .biz.de, .edu.uk.

Witryny mogą mieć różny charakter.
- Witryna firmowa prezentuje firmę, jej działalność, usługi lub produkty. Może to być serwis szkoły, ale także wizytówka zawierająca CV lub portfolio internauty.
- Portal wielotematyczny (horyzontalny) udostępnia informacje z różnych dziedzin, m.in. aktualne wiadomości z kraju i ze świata oraz prognozę pogody czy wyniki loterii. Oferuje konto pocztowe i przestrzeń dyskową, umożliwia dostosowanie treści do preferencji użytkownika.

- Portal tematyczny (wertykalny, wortal) oferuje dostęp do informacji z konkretnej dziedziny. Często umożliwia prowadzenie własnego konta, komentowanie, a w przypadku portalu fotograficznego – publikowanie i przechowywanie zdjęć.
- Serwis społecznościowy służy budowaniu sieci kontaktów, umożliwia dzielenie się informacjami przez osoby o wspólnych zainteresowaniach czy podobnym profilu zawodowym.
- Sklep online umożliwia kupowanie i sprzedawanie produktów oraz usług przez internet.
- Blog to prywatny lub otwarty e-dziennik zawierający odrębne wpisy – zarchiwizowane, otagowane (skategoryzowane) i udostępnione bądź nie do skomentowania. Poza typowym tekstowym blogiem można wyróżnić m.in. fotoblog (ze zdjęciami), wideoblog (z filmami), audioblog (z podcastami) czy linklog (z linkami i komentarzami do obcych treści).

Każda witryna ma określoną strukturę. Sposób podziału treści i funkcjonalności powinien być intuicyjny i nieskomplikowany. Kolejne strony należy łączyć za pomocą odpowiednich odnośników, a wyraźnie wyeksponowane przejrzyste menu umieszczać zawsze w tym samym miejscu.

Pierwszej stronie serwisu trzeba nadać nazwę **index** i właściwe rozszerzenie (np. dla HTML będzie to .htm lub .html). Dzięki temu będzie ona zawsze wczytywana po wpisaniu w przeglądarce adresu WWW.

Gotowe strony WWW oraz powiązane z nimi dokumenty publikuje się, tzn. wgrywa na wybrany serwer WWW za pomocą oprogramowania nazywanego klientem FTP (do najpopularniejszych tego typu aplikacji należą Total Commander i FileZilla).

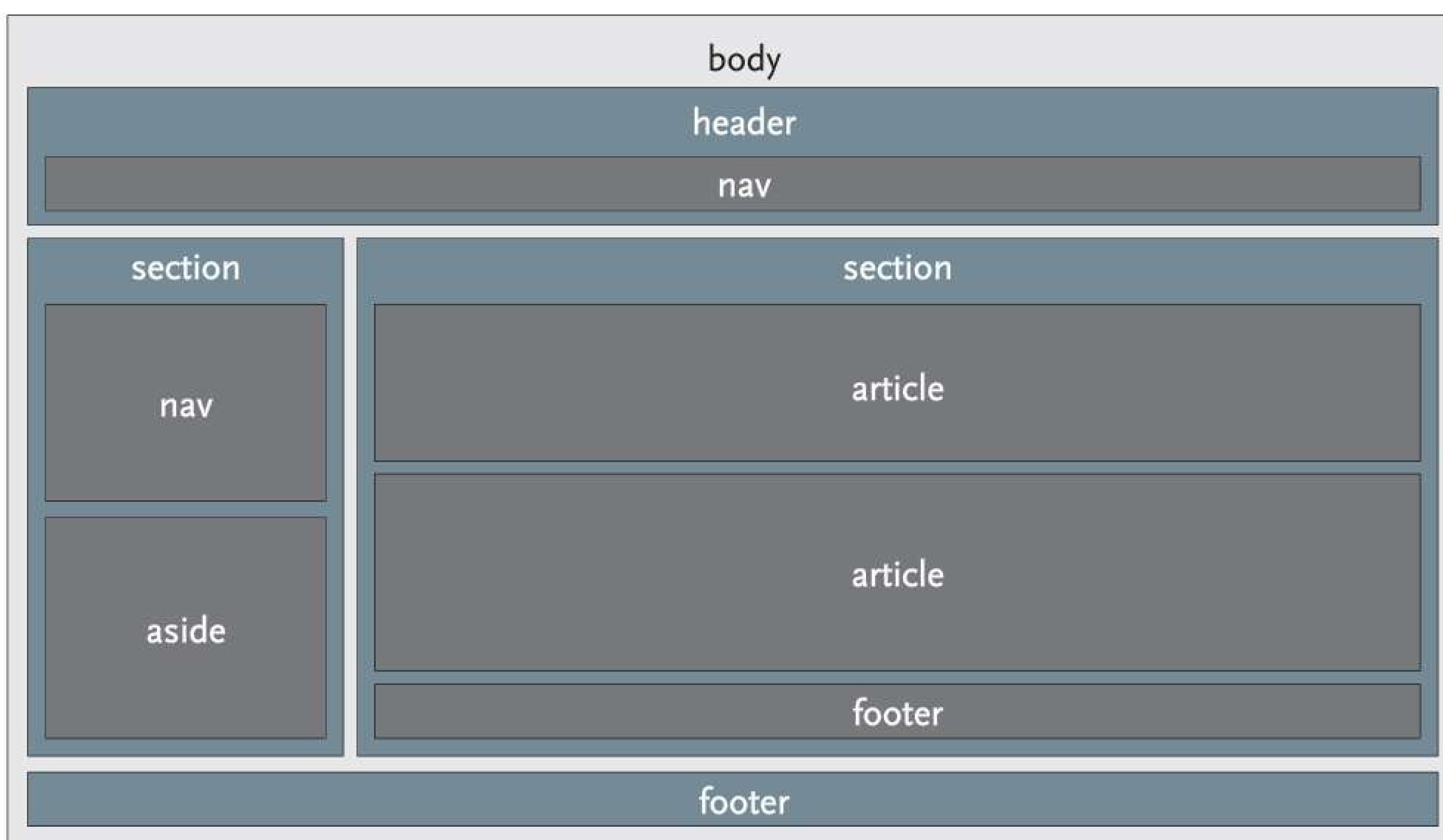
STRUKTURA LOGICZNA DOKUMENTU HTML

W specyfikacji HTML 5 wprowadzono znaczniki logiczne dotyczące organizacji dokumentu, które pozwalają wydzielić stałe obszary strony WWW i przypisać im konkretne role. Niektóre z nich już znasz.

Znacznik	Opis
`header`	obszar przeznaczony na nagłówek
`nav`	obszar przeznaczony na elementy nawigacyjne – menu
`article`	obszar zawierający oddzielną, niezależną część dokumentu
`section`	separator oddzielający odrębne bloki (sekcje) dokumentu
`aside`	obszar przeznaczony do umieszczania elementów dodatkowych, np. reklam
`footer`	obszar przeznaczony na stopkę dokumentu lub sekcji

Tab. 1. Znaczniki dotyczące organizacji dokumentu w specyfikacji HTML5

Opisane w tabeli elementy można rozmieścić np. w poniższy sposób.



Rys. 1. Przykładowa struktura logiczna dokumentu HTML

► Przejrzyj kody swoich dokumentów HTML (**ja2.htm**, **mojehobby.htm**, **galeria.htm**) i zgodnie z założeniami przyjętymi w języku HTML5 zmodyfikuj je za pomocą znaczników **`header`**, **`nav`**, **`article`**, **`section`** oraz **`footer`**. W stopce możesz umieścić swoje imię i nazwisko lub pseudonim, datę powstania witryny, notkę o zastrzeżeniu praw autorskich, a także odnośniki – do powiązanych dokumentów, stron w ramach serwisu lub innych serwisów.

SYSTEM ODNOŚNIKÓW

Aby połączyć dokumenty w jedną witrynę lub odesłać użytkownika do innego serwisu, trzeba uzupełnić kod źródłowy o odpowiednie linki. Wprowadza się je za pomocą znacznika **`<a>`** z atrybutem **`href`** wskazującym adres URL dokumentu, który ma się pojawić po kliknięciu odnośnika.

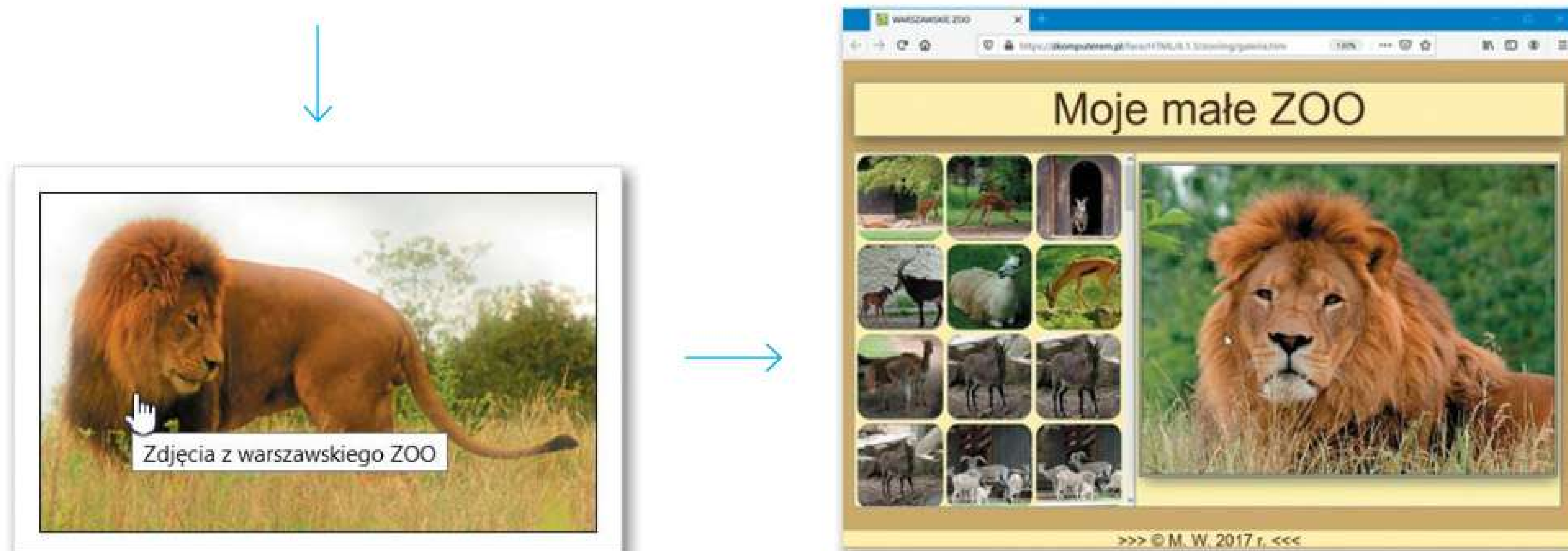
Odnośnik tekstowy „moje hobby”, po kliknięciu którego ma wgrywać się należąca do witryny strona **mojehobby.htm**, należy skonstruować następująco:



Rys. 2. Definicja odnośnika tekstowego i jej rezultat

Odnośnik graficzny, po kliknięciu którego powinna pojawić się galeria funkcjonująca poza witryną, należy zbudować w ten sposób:

```
<a href="http://zkomputerem.pl/face/HTML/8.1.3/zooimg/galeria.htm" title="Zdjęcia z warszawskiego ZOO">
 <img src="kotek.jpg" alt="Zdjęcia z warszawskiego ZOO" style="border: solid 1px black;">
</a>
```



Rys. 3. Definicja odnośnika graficznego i jej rezultat

Podczas tworzenia odnośników można wykorzystać również pseudoklasy.

Pseudoklasa	Opis	Przykładowy efekt
a:link {atrybut: wartość;}	ustalenie wyglądu odnośnika do dokumentu jeszcze nieodwiedzonego, np. kolor niebieski, podkreślenie	Regulamin
a:hover {atrybut: wartość;}	ustalenie wyglądu odnośnika po najechaniu na niego kursorem myszy, np. kolor niebieski, podkreślenie, żółte podświetlenie	Regulamin
a:active {atrybut: wartość;}	ustalenie wyglądu aktywnego (klikniętego) odnośnika, np. kolor czerwony, podkreślenie, żółte podświetlenie	Regulamin
a:visited {atrybut: wartość;}	ustalenie wyglądu odnośnika do dokumentu zapisanego w historii przeglądania, np. kolor fioletowy, podkreślenie	Regulamin

Tab. 2. Pseudoklasy, które można wykorzystać podczas tworzenia odnośników

Przy opisywaniu kilku stanów odnośnika obowiązują dwie reguły: **a:hover** występuje po **a:link** i **a:visited**, natomiast **a:active** – po **a:hover**.

► Zdefiniuj odnośnik tak, aby był czerwony, jeśli strona, do której kieruje, nie została jeszcze odwiedzona, zielony, jeśli strona została odwiedzona, pogrubiony, gdy najedzie się na niego kursorem, i pogrubiony niebieski, gdy kliknie się link. Zachowaj odpowiednią kolejność.

```
a:link {
  color: red;
}

a:visited {
  color: green;
}

a:hover {
 font-weight: bold;
}

a:active {
 font-weight: bold;
 color: blue;
 }
```

Rys. 4. Cztery stany odnośników

Znacznik <a> ma jeszcze jeden istotny atrybut – **`target`**. Wpływa on na sposób otwierania wywołanego dokumentu. Może przyjmować m.in. wartości:

- **`_blank`** – wywoływany dokument otworzy się w nowym oknie przeglądarki **`<a href="mojehobby.htm" target="_blank">moje hobby</a>`**;
- **`_self`** – wywoływany dokument otworzy się w bieżącym oknie przeglądarki **`<a href="mojehobby.htm" target="_self">moje hobby</a>`**. Wartość **`self`** jest domyślna, więc w kodzie może być pominięta.

► Zmień nazwę pliku **ja2.htm** na **index.htm**, a następnie połącz wszystkie dokumenty systemem odnośników, tak aby można było między nimi swobodnie nawigować.

- Umieść odnośniki w górnych częściach stron w jednowierszowym akapicie (znacznik **<p>**).
- Zadbaj o porządek. Utwórz folder **moja strona**, a w nim **img**. W folderze **moja strona** umieść wszystkie plik HTML, a w folderze **img** wszystkie pliki graficzne. Pamiętaj, aby w odwołaniach do plików podać właściwą ścieżkę dostępu. Odwołanie do pliku **obrazek.jpg** powinno wyglądać następująco: **`<img src="img/obrazek.jpg">`**.

Kursy projektowania stron WWW

Jeśli zainteresował cię temat tworzenia stron WWW, skorzystaj z bezpłatnych kursów oferowanych przez Akademię Khana (**https://pl.khanacademy.org**) *Wstęp do HTML/CSS: Tworzenie stron internetowych* oraz *HTML/JS: Tworzenie interaktywnych stron internetowych*. Pracę z kodem źródłowym ułatwią ci prezentowane na bieżąco wskazówki.

Aby w pełni korzystać z możliwości oferowanych przez serwis Akademii Khana, uzyskać wsparcie w nauce i śledzić swoje postępy, należy się zarejestrować.

ZADANIA

1. Zgodnie z instrukcją podaną przez nauczyciela skopiuj na szkolny serwer WWW pliki składowe swojej witryny i sprawdź w przeglądarce, czy wszystko właściwie działa. W razie konieczności otwórz odpowiedni plik z dysku twardego, wprowadź poprawki i ponownie wyślij dokument na serwer.

2. Otwórz kurs *Wstęp do HTML/CSS: Tworzenie stron internetowych* w Akademii Khana i znajdź temat *Dalsza nauka*. Zapoznaj się z informacjami dotyczącymi projektu strony internetowej oraz walidacji HTML, a następnie sprawdź swoją wiedzę.

1.6 Prawo w internecie

DOWIESZ SIĘ,

- co przepisy prawa autorskiego mówią na temat ochrony twórczości i wizerunku,
- czym jest wolne oprogramowanie.

Ustawa z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych mówi, że nie wolno korzystać z wytworów innych osób bez ich zgody, a w niektórych przypadkach bez uiszczania wynagrodzenia. Co mówią przepisy na temat kopiowania zawartości witryn oraz ochrony oprogramowania i wizerunku?

STRONY INTERNETOWE

Witryna lub pojedyncza strona internetowa podlega ochronie prawa autorskiego, jeśli można o niej powiedzieć, że jest utworem, czyli stanowi efekt pracy twórczej o indywidualnym charakterze, zawiera stworzone specjalnie na potrzeby serwisu elementy graficzne, teksty, fotografie, filmy, muzykę, nietypowy układ menu itp.

Warto pamiętać, że prawa autorskie narusza zarówno skopiowanie całej struktury witryny wraz z prezentowanymi materiałami oraz systemem powiązań kolejnych stron, jak i powielenie szaty graficznej lub wybranych elementów czy usunięcie znaku wodnego z udostępnionych bezpłatnie zdjęć i opublikowanie ich na własnej stronie.

W przypadku naruszenia praw autorskich do witryny lub strony WWW autor może domagać się m.in. usunięcia ze strony ukradzionych elementów (utworów) lub uiszczenia za nie opłaty. Jeśli chcesz zatem dodać do swojej strony jakieś elementy graficzne, zanim to zrobisz, wyślij e-mail z prośbą o pozwolenie do ich autorów lub właścicieli serwisów, w których zostały zamieszczone. A co jeśli nie znasz adresu e-mail autora lub właściciela? Sprawdź, czy na stronie zawierającej interesujący cię element zamieszczono informację mówiącą, czy możesz go wykorzystać, i jeśli tak, to w jaki sposób. Czasem wystarczy podać dane autora obrazu lub adresu strony, z której został pobrany. Najlepszym jednak rozwiązaniem jest korzystanie z elementów graficznych dostępnych na stronach zawierających darmowe zasoby graficzne. W internecie istnieje bardzo dużo takich stron. Jedną z nich jest **https://pixabay.com/pl/**.

► Czy wiesz, jak określa się fakt przypisania sobie autorstwa cudzego utworu? Jeśli nie wiesz, to poszukaj informacji na ten temat w internecie. Wpisz w pole wyszukiwania tekst: „przypisanie sobie autorstwa cudzego utworu”.

DOZWOLONY UŻYTEK PRYWATNY

Zastanawiasz się, czy możesz kopiować utwory muzyczne dostępne w internecie? Zgodnie z tak zwanym dozwolonym użytkiem prywatnym możesz skopiować na własne potrzeby ulubioną piosenkę dostępną w internecie, ale pod warunkiem że utwór ten został upubliczniony za zgodą wykonawcy. Nie możesz jednak z odtwarzania tego utworu czerpać korzyści finansowych. Jeśli utwór został nagrany, np. w czasie koncertu, i rozpowszechniony w sieci bez zgody wykonawcy, to kopiowanie takiego utworu nie jest objęte dozwolonym użytkiem prywatnym, czyli jest niedozwolone.

OPROGRAMOWANIE

Ustawa o prawie autorskim i prawach pokrewnych w szczególny sposób chroni programy komputerowe. Zgodnie z zapisem w rozdziale 7 ustawy ochrona obejmuje formę programu, czyli m.in. informacje zawarte w kodzie źródłowym i w opisie procedur, które stanowią elementy twórcze.

Jak już wiesz, legalne korzystanie z programu komputerowego jest możliwe po zawarciu umowy licencyjnej – zwykle poprzez otwarcie opakowania z nośnikiem (to zagadnienie było omawiane w klasie 7). Licencja informuje, na jakich warunkach licencjobiorca może korzystać z programu, np. niekomercyjnie, na maksymalnie dwóch stanowiskach komputerowych, przez dwa lata.

Należy pamiętać, że w przypadku programów komputerowych nie obowiązuje dozwolony użytek prywatny, co oznacza, że nie możesz udostępniać nikomu oryginalnego oprogramowania. Zapalony gracz, który chciałby zamieszczać na swojej stronie zrzuty ekranowe z ulubionych gier albo zapis całych rozgrywek lub dodawać do nich swoje opisy, powinien za każdym razem sprawdzać, co napisano na ten temat w licencji. To samo dotyczy zamieszczania na stronie darmowego programu do ściągnięcia – w przypadku licencji shareware trzeba się upewnić, że zezwala ona na dalszą dystrybucję. Bez pytania o zgodę można rozpowszechniać tylko programy typu freeware i wolne oprogramowanie.

► Załóżmy, że twoim hobby jest grafika komputerowa. W związku z tym używasz wielu programów graficznych, ale najbardziej przypadł ci do gustu GIMP. Nawet prowadzisz stronę internetową poświęconą temu oprogramowaniu. Chcesz na niej udostępnić swoim czytelnikom ulubiony program – GIMP. Jak myślisz, czy możesz to zrobić bez niczyjej zgody?

Wolne oprogramowanie (ang. *free software;* czytaj: fri softłer) jest chronione bezpłatną licencją i powszechnie dostępne; można je legalnie pobierać z internetu. Do takich programów należą np. system operacyjny Linux, pakiety Apache OpenOffice (czytaj: apacz ołpenofis) oraz LibreOffice czy edytor nagrań dźwiękowych Audacity (czytaj: odasiti).

W trakcie instalacji oraz pierwszego uruchomienia programu trzeba wyrazić zgodę na warunki licencji GNU GPL (ang. *GNU General Public License;* czytaj: dżenrel pablik lajsynz – powszechna licencja GNU) lub GNU LGPL (ang. *GNU Lesser General Public*

License; czytaj: lese dżenrel pablik lajsynz – mniejsza powszechna licencja GNU). Pozwalają one na praktycznie dowolne bezpłatne użytkowanie programu, nie wykluczają jednak, że pewne jego fragmenty mogą stanowić czyjąś własność i być chronione przed przekształcaniem.

Cztery rodzaje wolności użytkowników programu udostępnionego na wolnej licencji

- Wolność uruchamiania programu w dowolnym celu.
- Wolność analizowania, jak działa program, i dostosowywania go do swoich potrzeb.
- Wolność rozpowszechniania kopii programu, aby pomóc tym, którzy go potrzebują.
- Wolność udoskonalania programu i publicznego rozpowszechniania własnych ulepszeń, by wszyscy mogli z nich korzystać.

► Ruch na rzecz wolnego oprogramowania uruchomił aktywność wielu grup, które stawiają sobie za cel bezpłatną prezentację w internecie różnych materiałów, np. lektur szkolnych czy podręczników. Znajdź informacje na temat Richarda M. Stallmana (czytaj: riczarda em stolmena). Dowiedz się, kim jest i jaka idea przyświeca jego działalności.

► Wejdź na stronę **codewith.mu** i przejdź do sekcji **Download**. Znajdziesz tam program o nazwie Mu (czytaj: mju). Jest to prosty edytor tekstu na licencji GNU. Pobierz wersję właściwą dla twojego systemu operacyjnego i zainstaluj ją na swoim komputerze. Z aplikacji tej będziesz korzystać w czasie lekcji poświęconych programowaniu.

Instalacja oprogramowania

Instalowanie oprogramowania często wymaga uprawnień administratora komputera lub sieci. Nie próbuj instalować nowych programów w pracowni szkolnej bez zgody nauczyciela. Instalację na komputerze domowym uzgodnij z rodzicami lub opiekunami.

Coraz częściej twórcy wykorzystują licencje z grupy Creative Commons (CC). Ich treści można znaleźć na stronie **creativecommons.pl**.

OCHRONA WIZERUNKU I BEZPIECZEŃSTWO W SIECI

W rozdziale 10 *Ustawa o prawie autorskim i prawach pokrewnych* skupia się m.in. na ochronie wizerunku. Zgodnie z przepisami rozpowszechnianie wizerunku (np. zdjęcia, filmu) wymaga zezwolenia osoby na nim przedstawionej, chyba że chodzi o wizerunek osoby powszechnie znanej, zaprezentowanej w związku z pełnieniem przez nią funkcji publicznych, np. politycznych, lub jeśli dana osoba stanowi szczegół większej całości, np. publiczności podczas koncertu.

Postępowanie naruszające prawo

- Publikowanie zdjęcia przedstawiającego inną osobę bez jej zgody.
- Publikowanie tekstu obrażającego inną osobę.
- Publikowanie adresu lub telefonu innej osoby bez jej zgody.

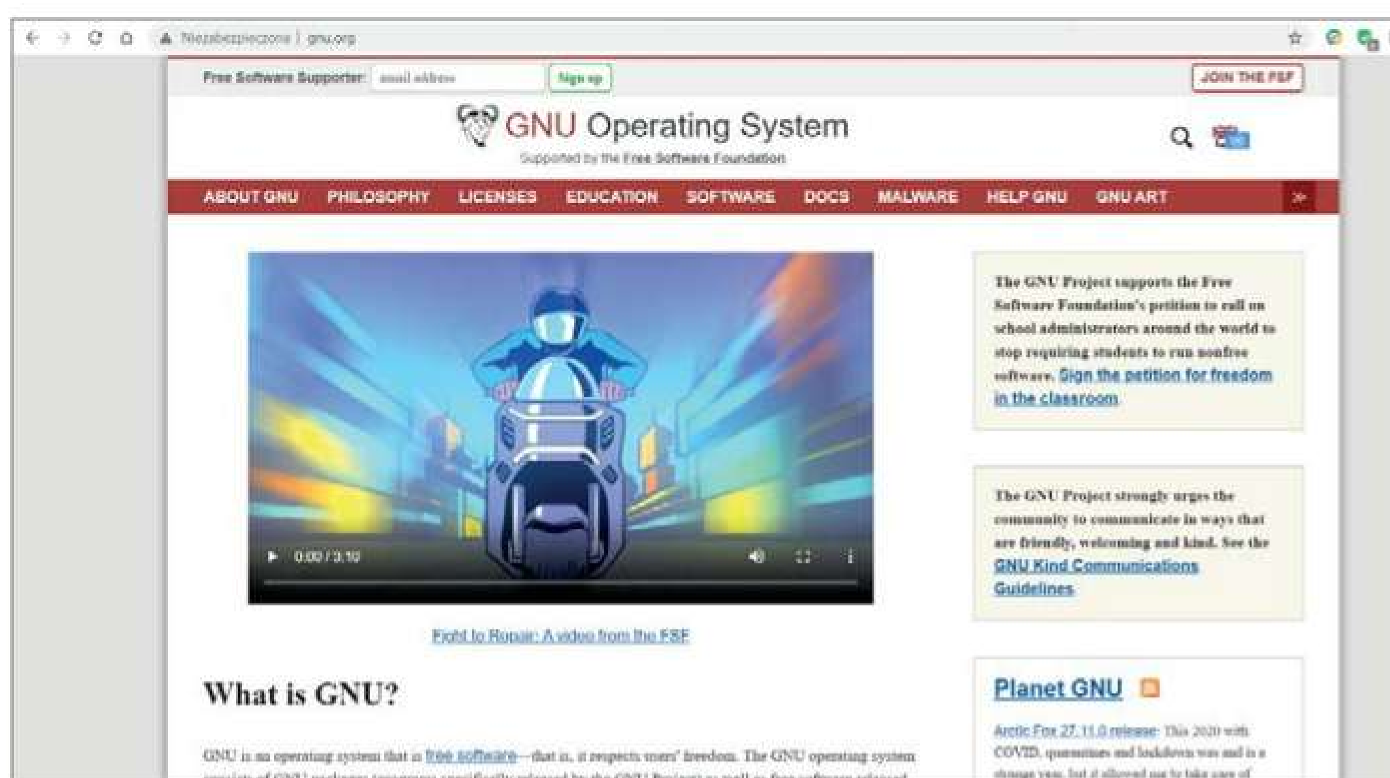
Pamiętaj, że nawet najlepsze prawo nie wystarczy, jeśli nie będziesz przestrzegać zasad bezpieczeństwa w sieci:

- zachowaj dyskrecję – pamiętaj, że informacje, które zamieszczasz w sieci, stają się publiczne i mogą zacząć żyć swoim życiem; zanim coś napiszesz albo dodasz jakieś zdjęcie czy film, zastanów się, jak może zareagować na to twoje otoczenie;
- zachowaj dystans wobec osób poznanych w sieci – nie podawaj obcym swoich danych osobowych (imienia i nazwiska, adresu e-mail, adresu domowego i numeru telefonu); nie wysyłaj obcym swoich zdjęć; pamiętaj, że tak naprawdę nie wiesz, kim jest twój rozmówca;
- zadbaj o bezpieczeństwo danych przechowywanych na komputerze – stosuj bezpieczne hasło, używaj programu antywirusowego, ustaw odpowiednie zabezpieczenia w przeglądarce, nie otwieraj i nie przesyłaj dalej e-maili czy wiadomości z komunikatora, jeśli nie znasz nadawcy.

ZADANIA

1. Znajdź w internecie *Ustawę z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych* i jej omówienia. Na ich podstawie utwórz złożoną z dwóch stron prostą witrynę WWW prezentującą najważniejsze zagadnienia dotyczące tematu *Prawo autorskie – ochrona twórczości i wizerunku*.

2. Na podstawie stron WWW poświęconych wolnemu oprogramowaniu opracuj prezentację *Wolne oprogramowanie*.



Na co zwracać uwagę w zapisie dokumentu HTML?

- Należy stosować podstawową strukturę dokumentu, zdefiniowaną przez World Wide Web Consortium.
- Dobrym zwyczajem jest tworzenie struktury logicznej dokumentu przez stosowanie znaczników, takich jak: **`header`**, **`nav`**, **`article`**, **`section`**, **`aside`** oraz **`footer`**.
- Aby zapewnić poprawność wyświetlania znaków specjalnych, w sekcji **`head`** należy umieszczać deklarację sposobu kodowania znaków (UTF-8).
- Znaczniki zamykające należy zapisywać w odwrotnej kolejności niż otwierające. Obowiązuje zasada, że jeśli połączysz klamrami pary znaczników otwierających i zamykających, to klamry nie mogą się krzyżować.

```
<p style="font-size: 16pt;"><em><strong>Widomość dnia:</strong> Ala będzie weterynarzem!</em></p>
```

- Znaczniki, ich atrybuty i wartości należy zapisywać małymi literami, a w przypadku odwołań do innych plików trzeba stosować oryginalny zapis ich nazw (wielkie i małe litery).
- Pojedyncze litery na końcu wiersza tekstu należy przenosić do następnego wiersza przez wstawienie znaku specjalnego **` `** między taką literę a następne słowo.



Jakie są zalety zewnętrznych stylów CSS?

Style zewnętrzne ułatwiają zarządzanie całą witryną. Wystarczy zdefiniować styl, np. dla akapitu, aby we wszystkich dokumentach witryny wszystkie akapity miały identyczny format. W razie konieczności modyfikacji wyglądu jakiegoś elementu wystarczy zmienić instrukcję w jednym miejscu arkusza zewnętrznego, a zmiany automatycznie pojawią się w pozostałych dokumentach witryny.

Jak można stworzyć przycisk animowany?

Do tworzenia przycisków animowanych często wykorzystuje się pseudoklasę **`:hover`**. Zmianę wyglądu przycisku można też wywołać w reakcji na zachodzące zdarzenia, takie jak: **`onclick`**, **`onmouseover`** i **`onmouseout`**.





2 Lekcje z Pythonem



W 1990 r. Holender Guido van Rossum (czytaj: głido wan rosum) szukał nazwy dla stworzonego przez siebie języka programowania. Ostatecznie zdecydował się na słowo *python* (czytaj: pajton), w nawiązaniu do słynącego z absurdalnego humoru brytyjskiego serialu komediowego *Latający Cyrk Monty Pythona*. Python ma wielbicieli na całym świecie. Tim Peters (czytaj: tim piters) napisał na temat tego języka wiersz *The Zen of Python* (Zen Pythona). Oto jego fragment w oryginale i tłumaczeniu:

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
[...]

Piękne jest lepsze niż brzydkie.
Jednoznaczne jest lepsze niż domniemane.
Proste jest lepsze niż złożone.
Złożone jest lepsze niż skomplikowane.
[...]



2.1 Rysuj z żółwiem

DOWIESZ SIĘ, JAK

- pisać proste polecenia w języku Python,
- rysować z wykorzystaniem modułu **`turtle`**,
- wykorzystać pętlę **`for`** do iteracji.

Umiesz programować wizualnie – tworzyć mniej i bardziej skomplikowane historyjki, gry oraz programy wykorzystujące matematyczne algorytmy w Scratchu i Blockly. Czas na pierwszą lekcję programowania tekstowego. Zaczniesz od rysowania.

PYTHON I EDYTOR MU

Python to język programowania wysokiego poziomu, czyli taki, którego składnia i kod są zrozumiałe przede wszystkim dla człowieka. Korzystają z niego zarówno początkujący programiści, jak i osoby zawodowo zajmujące się programowaniem.

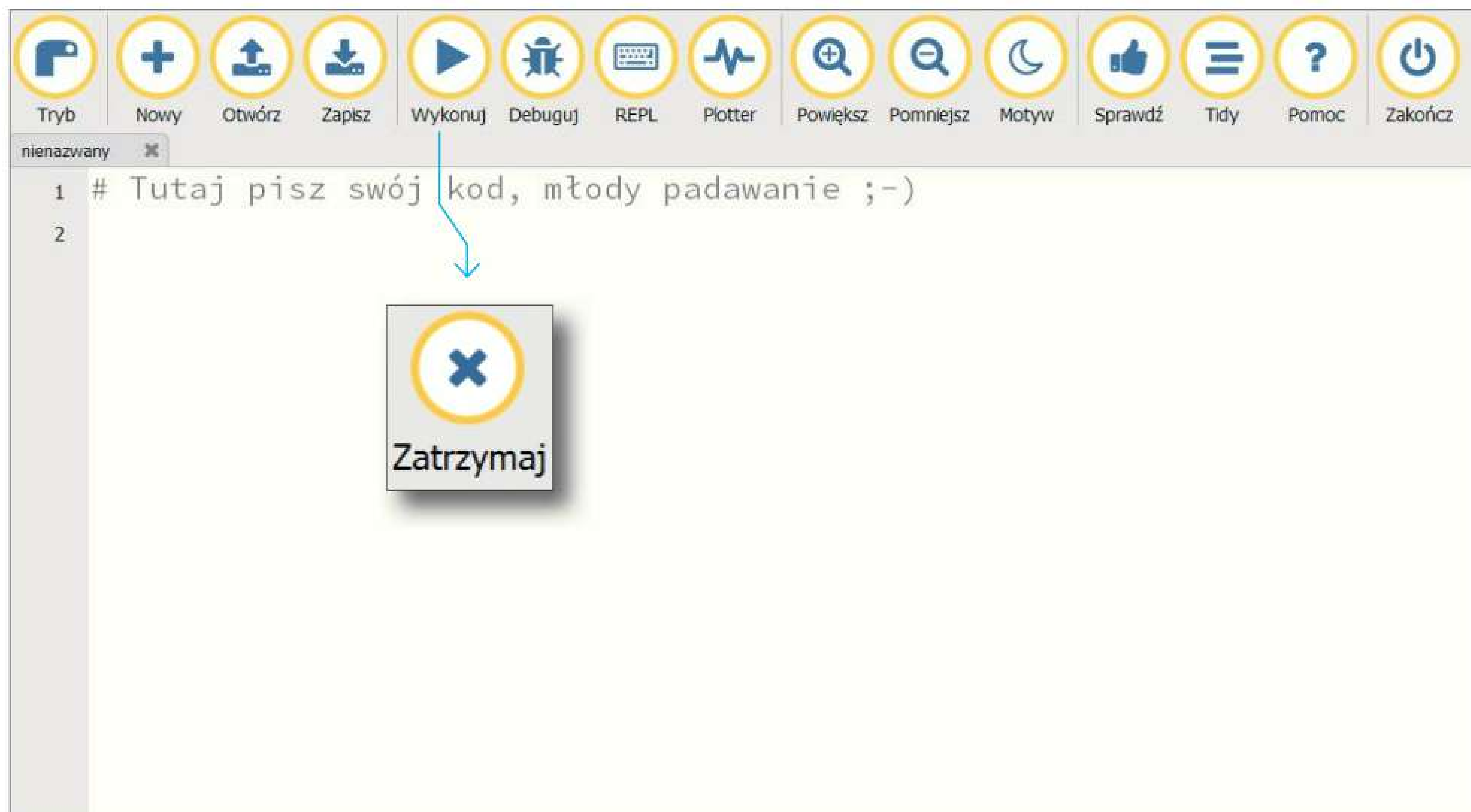
> **Składnia**
>
> W informatyce składnia oznacza zbiór symboli dostępnych w danym języku oraz zbiór reguł dotyczący łączenia tych symboli w większe struktury.

► Uruchom edytor Mu. Gdy robisz to po raz pierwszy, musisz wybrać tryb pracy w Pythonie w wersji 3.



Rys. 1. Wybór trybu pracy edytora Mu

► Przeanalizuj okno edytora – za co odpowiadają kolejne przyciski?



Rys. 2. Okno edytora Mu

Główne przyciski na tym etapie nauki to **Nowy**, **Otwórz** i **Zapisz**, a także przycisk **Wykonuj**, służący do uruchomienia napisanego programu i zmieniający się w przycisk **Zatrzymaj** – wykonywania programu (np. w celu ponownej edycji). Przyciski z przedostatniej grupy umożliwiają modyfikację edytora – zmniejszenie (–) lub powiększenie (+) czcionki i zmianę kolorów w celu zwiększenia kontrastu (**Motyw**). Przycisk **Sprawdź** uruchamia wyszukiwanie błędów, przy czym część informacji jest podawana w języku polskim, a część w języku angielskim.



Rys. 3. Przykładowe informacje na temat błędów w kodzie w edytorze Mu

RYSOWANIE W PYTHONIE

Grafika żółwia stanowi podstawę języka programowania Logo, który początkowo służył do sterowania robotem wyglądem przypominającym żółwia. Obecnie jest wykorzystywana w wielu narzędziach i środowiskach do nauki programowania, choć niekoniecznie pod swoją pierwotną nazwą. W Scratchu odpowiada za nią rozszerzenie **Pióro**, w Pythonie – moduł `turtle` (czytaj: tartl; żółw), który trzeba importować za pomocą polecenia `from turtle import *`.

Moduły w Pythonie

Moduł to plik zawierający zestawienie pewnych definicji i instrukcji do wykorzystania w innych programach w języku Python, np. moduł **`turtle`** udostępnia podstawowe operacje grafiki, a moduł **`math`** – funkcje matematyczne. Alfabetyczny spis standardowych modułów Pythona można znaleźć na stronie **https://docs.python.org/3/**.

- Przeanalizuj instrukcje grafiki żółwia zestawione w poniższej tabeli – zwróć uwagę na brak spacji przed nawiasem oraz obecność lub nieobecność parametru (np. **`alfa`**, **`n`**) w nawiasie.

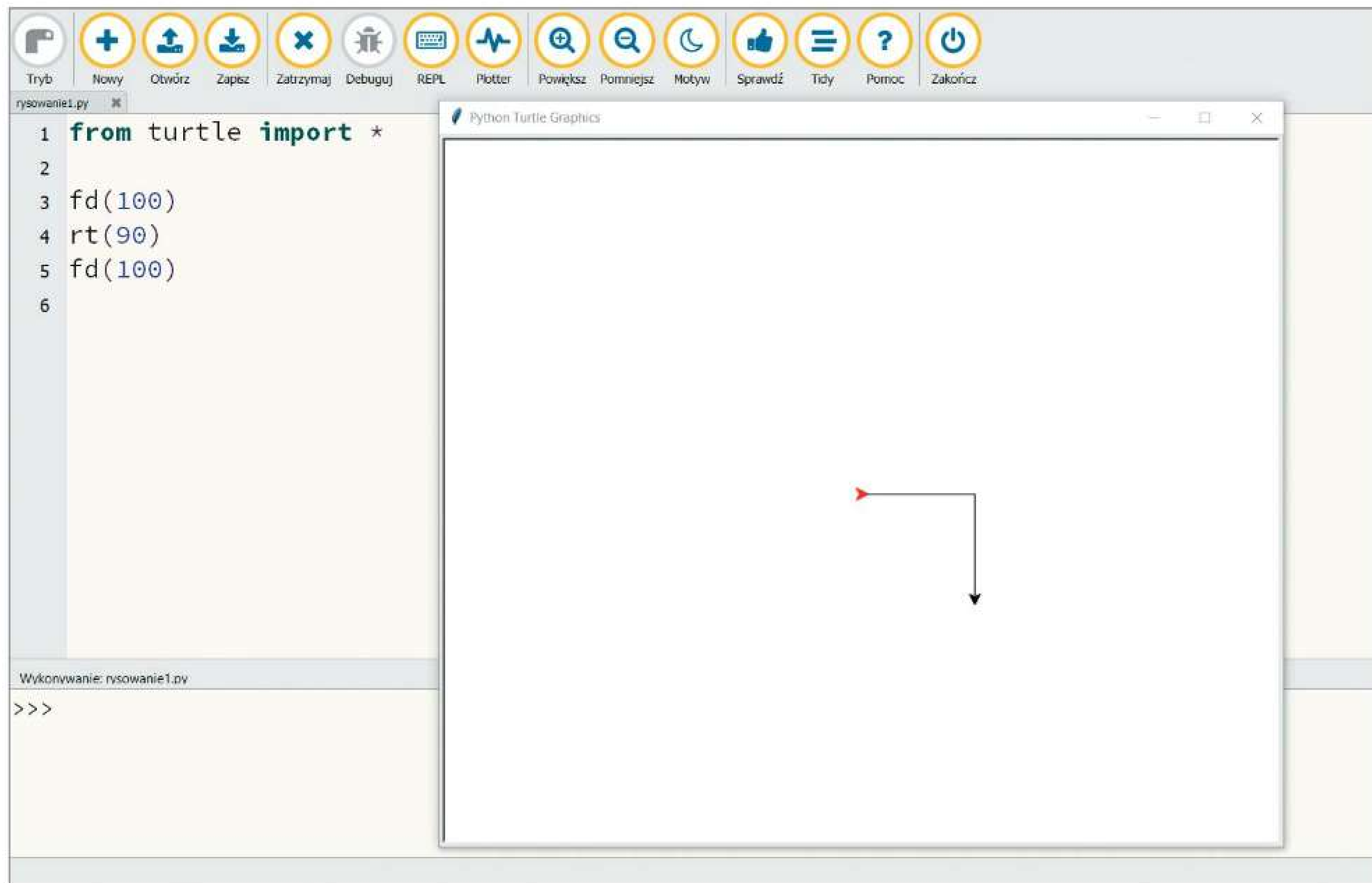
Polecenie	Znaczenie	Opis działania
`fd(n)`	*forward* – do przodu	Żółw przesuwa się bez zmiany kierunku o **`n`** kroków.
`bk(n)`	*backward* – wstecz	Żółw przesuwa się wstecz o **`n`** kroków.
`rt(alfa)`	*right* – prawo	Żółw obraca się o podany kąt **`alfa`** w prawo.
`lt(alfa)`	*left* – lewo	Żółw obraca się o podany kąt **`alfa`** w lewo.
`pu()`	*pen up* – podnieś pisak	Żółw podnosi pisak (nie rysuje podczas przemieszczania się).
`pd()`	*pen down* – opuść pisak	Żółw opuszcza pisak (rysuje podczas przemieszczania się).

Tab. 1. Wybrane polecenia grafiki żółwia

Należy pamiętać, że żółw zawsze startuje ze środka ekranu z opuszczonym pisakiem i jest skierowany w prawą stronę.

Sprawdź, jak rysuje żółw – niech zrobi do przodu 100 kroków, potem skręci w prawo o 90° i znów zrobi do przodu 100 kroków.

- Zaimportuj moduł **`turtle`**.
- Wydaj kolejne polecenia żółwiowi – możesz umieścić wszystkie instrukcje w jednym wierszu i oddzielić je średnikami albo każdą instrukcję zapisać w osobnym wierszu.
- Uruchom program – kliknij przycisk **Wykonaj**.
  - Pierwsze uruchomienie programu zawsze powoduje otwarcie okna zapisu pliku, każde kolejne uruchomienie – automatyczny zapis bieżącego stanu.
  - Po zapisaniu projektu otworzy się okno **Python Turtle Graphics**, w którym żółw w postaci strzałki wykona polecenia.



Rys. 4. Rysowanie z wykorzystaniem modułu **`turtle`** – dodana do zrzutu czerwona strzałka wskazuje pozycję wyjściową żółwia

ITERACJA, CZYLI POWTARZANIE

Do kilkukrotnego powtórzenia tej samej instrukcji należy wykorzystać pętlę **`for`**.

```
for <zmienna sterująca pętlą> in <sekwencja>:
    <instrukcje>
```

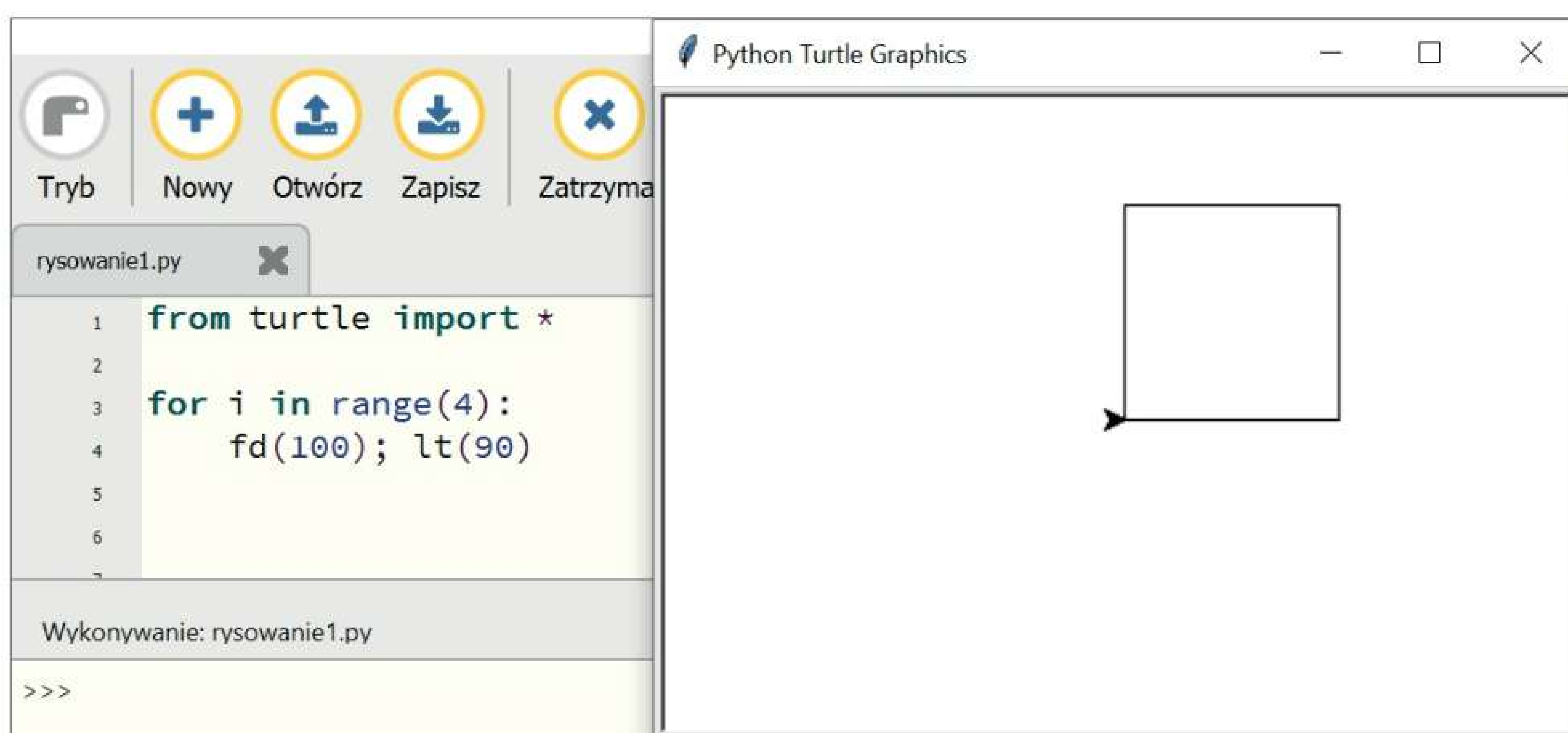
Rys. 5. Składnia pętli **`for`**

Po naciśnięciu klawisza **Enter** dwukropek na końcu pierwszego wiersza pętli generuje wcięcie w kolejnym wierszu – jeśli zabraknie dwukropka, wcięcie się nie zrobi, a podczas uruchamiania programu pojawi się błąd.

Zmienną, która steruje pętlą, zwykle oznacza się literą **`i`**. Jako sekwencji używa się funkcji **`range()`** – może ona mieć od jednego do trzech parametrów. Jeden parametr informuje o tym, ile razy pętla zostanie wykonana, np. **`range(3)`** oznacza trzy powtórzenia podanych instrukcji.

Zastosuj pętlę **`for`** do narysowania kwadratu.

- Zaimportuj moduł **`turtle`**.
- Zdefiniuj pętlę – w tym przypadku instrukcja „naprzód o 100 i obrót w lewo o 90" powinna wykonać się cztery razy. Pamiętaj o dwukropku.



Rys. 6. Przykładowy program realizujący rysowanie kwadratu

RYSOWANIE Z KOLORAMI

Kolorowe rysowanie wymaga zmiany koloru pisaka i zamalowania figury. Wykorzystuje się do tego poniższe polecenia:

- **`pencolor("green")`** – ustalenie koloru pisaka (tutaj zielony);
- **`fillcolor("blue")`** – ustalenie koloru zamalowania (tutaj niebieski);
- **`begin_fill()`** – początek zamalowywania figury;
- **`end_fill()`** – koniec zamalowywania figury.

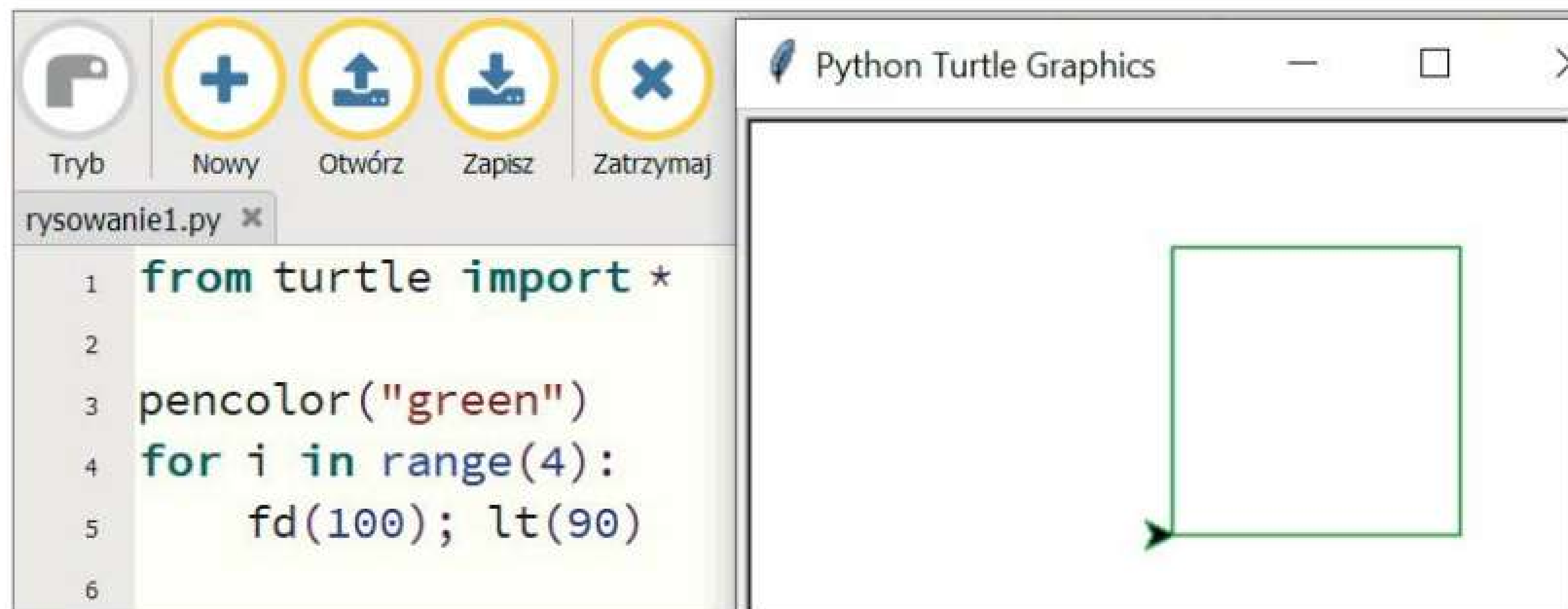
Nazwy kolorów należy podawać w języku angielskim, przy czym oprócz podstawowych kolorów (red, green, blue, yellow, black) można odwoływać się też do nietypowych odcieni, takich jak olivedrab, tomato, gold.

nazwa	kolor	nazwa	kolor	nazwa	kolor	nazwa	kolor
black		darkcyan		gray		red	
blue		dark-magenta		green		tomato	
cyan		darkred		magenta		white	
darkblue		gold		olivedrab		yellow	

Tab. 2. Wybrane nazwy kolorów w grafice żółwia

Narysuj kwadrat w kolorze zielonym.

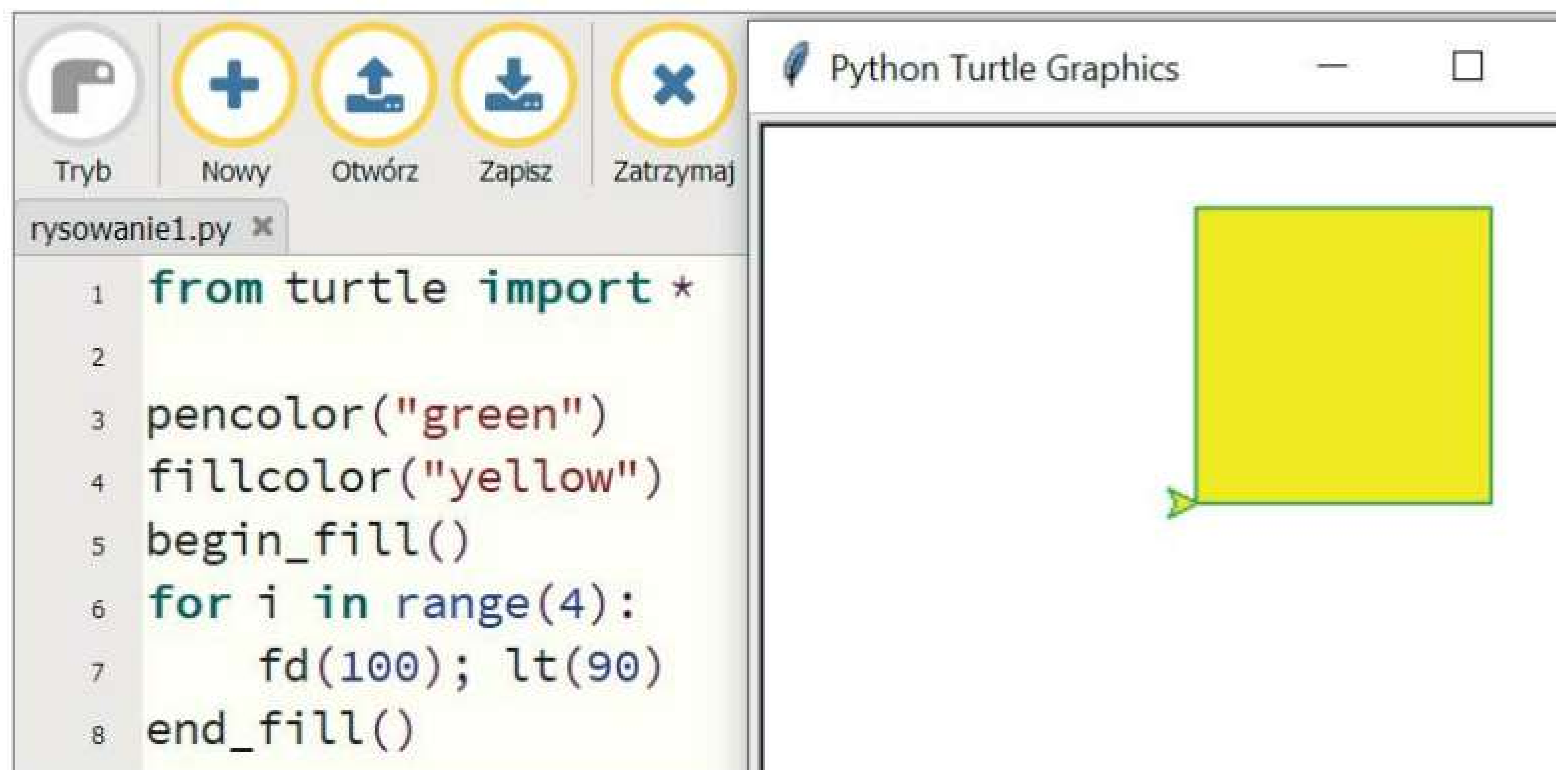
▶ Do poprzedniego skryptu dodaj polecenie ustawienia koloru pisaka. Umieść je przed pętlą odpowiedzialną za rysowanie.



Rys. 7. Przykładowy program realizujący rysowanie kwadratu zielonym pisakiem

Ponownie zmodyfikuj program – wypełnij kwadrat kolorem żółtym.

▶ Dodaj do skryptu trzy polecenia – ustawienia koloru zamalowania oraz początku i końca zamalowywania. Dwa pierwsze polecenia powinny znaleźć się przed pętlą, ostatnie polecenie trzeba umieścić po zakończeniu pętli, koniecznie bez wcięcia.

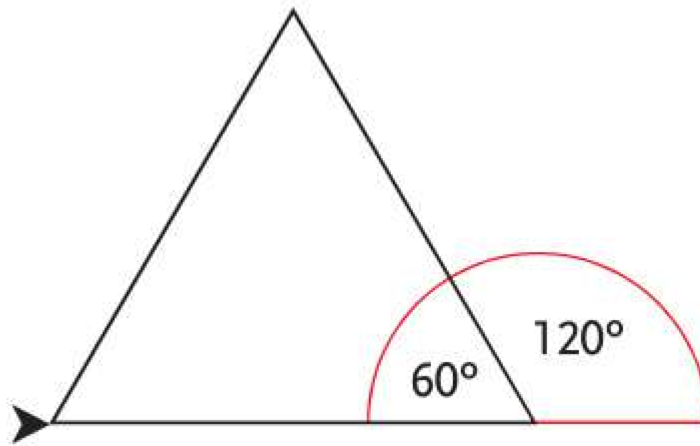


Rys. 8. Przykładowy program realizujący rysowanie kwadratu o zielonej krawędzi i żółtym wypełnieniu

Zmień program tak, aby zamiast kwadratu został narysowany trójkąt równoboczny.

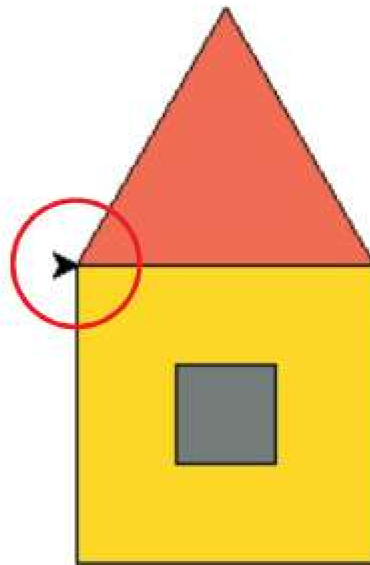
▶ Zmień wartość parametru funkcji **`range`** – tym razem instrukcję trzeba powtórzyć trzy razy.

▶ Zmień miarę kąta, o jaki musi obrócić się żółw po narysowaniu boku trójkąta – wszystkie kąty trójkąta równobocznego mają miarę 60°, zatem kąt obrotu żółwia wynosi 120°.



Rys. 9. Oczekiwany efekt skryptu rysowania trójkąta równobocznego – na czerwono zaznaczono kąty

A jak narysować domek o fasadzie szerokości 120 i dachu w kształcie trójkąta równobocznego, taki jak na rysunku poniżej?



Rys. 10. Domek – strzałką zaznaczono pozycję wyjściową żółwia

- Przeanalizuj problem: żółw stoi w miejscu, w którym zaczyna rysować; rysowanie fasady i dachu zacznie w tym samym miejscu, przy czym fasadę narysuje jako kwadrat, obracając się w prawo, a dach – jako trójkąt, obracając się w lewo. Aby narysować okno, trzeba podnieść pisak, przenieść go w nowe miejsce i opuścić.
- Zaimportuj moduł **`turtle`**.
- Narysuj fasadę domku – zamalowany kwadrat. Przetestuj ten fragment skryptu – sprawdź, czy rysunek jest poprawny i czy żółw stoi tak jak przed rysowaniem.

```
from turtle import *
# fasada domku
fillcolor("gold")
begin_fill()
for i in range(4):
    fd(120); rt(90)
end_fill()
```

Rys. 11. Przykładowy skrypt rysowania fasady

#

Wiersz rozpoczynający się od znaku **#** Python traktuje jako komentarz i pomija przy sprawdzaniu lub wykonywaniu programu.

► Narysuj dach – zamalowany trójkąt równoboczny. Przetestuj ten fragment – sprawdź, czy rysunek jest poprawny i czy żółw stoi tak jak przed rysowaniem.

```
# dach
fillcolor("tomato")
begin_fill()
for i in range(3):
    fd(120); lt(120)
end_fill()
```

Rys. 12. Przykładowy skrypt rysowania dachu

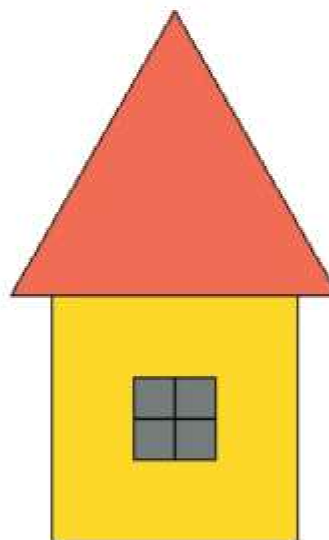
► Narysuj okno – podnieś pisak, przemieść żółwia w miejsce, gdzie możesz rozpocząć rysowanie okna, opuść pisak i narysuj zamalowany kwadrat. Przetestuj ten fragment skryptu – sprawdź, czy rysunek jest poprawny.

```
# okno
pu(); fd(40); rt(90); fd(40); pd()
fillcolor("gray")
begin_fill()
for i in range(4):
    fd(40); lt(90)
end_fill()
```

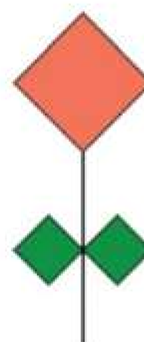
Rys. 13. Przykładowy skrypt rysowania okna

ZADANIA

1. Zmodyfikuj skrypt rysowania domku tak, aby osiągnąć taki efekt, jak na rysunku poniżej.



2. Narysuj kwiatek złożony z kwadratów.



2.2 Fantazyjne posadzki

DOWIESZ SIĘ, JAK

- definiować funkcje,
- korzystać z funkcji pomocniczych,
- tworzyć powtarzające się wzory.

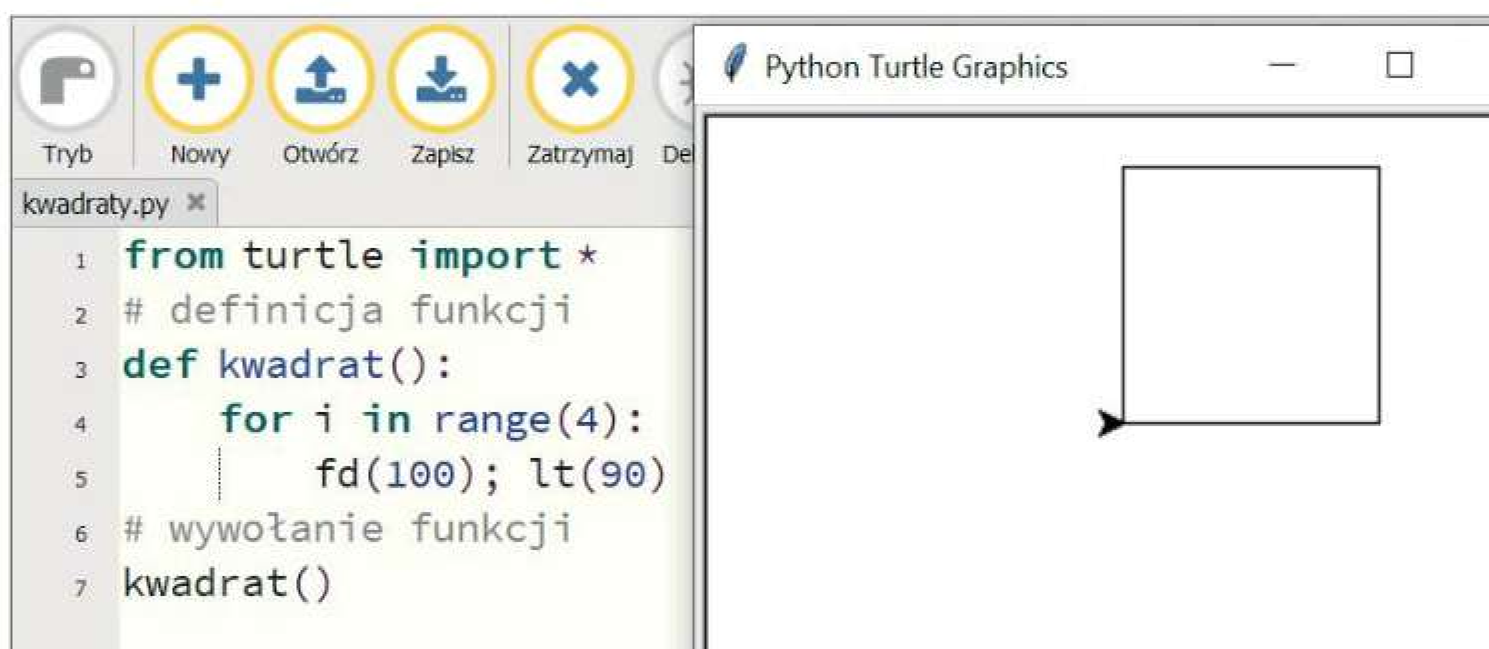
Aby program był bardziej zwięzły i przejrzysty, a fragmenty kodu stanowiące logiczną całość dało się wykorzystać wielokrotnie, warto tworzyć funkcje.

DEFINIOWANIE FUNKCJI BEZ PARAMETRU

Definicję funkcji rozpoczyna się słowem **`def`**, następnie podaje się nazwę funkcji, a za nią wpisuje pusty nawias i dwukropek (po naciśnięciu klawisza **Enter** w następnym wierszu powstanie wcięcie, co jest kluczowe do poprawnego zapisu funkcji). Kolejne wiersze definicji zawierają polecenia do wykonania. Aby wywołać funkcję, wystarczy podać jej nazwę wraz z pustym nawiasem.

Zdefiniuj funkcję **`kwadrat()`** odpowiadającą za rysowanie kwadratu o boku 100.

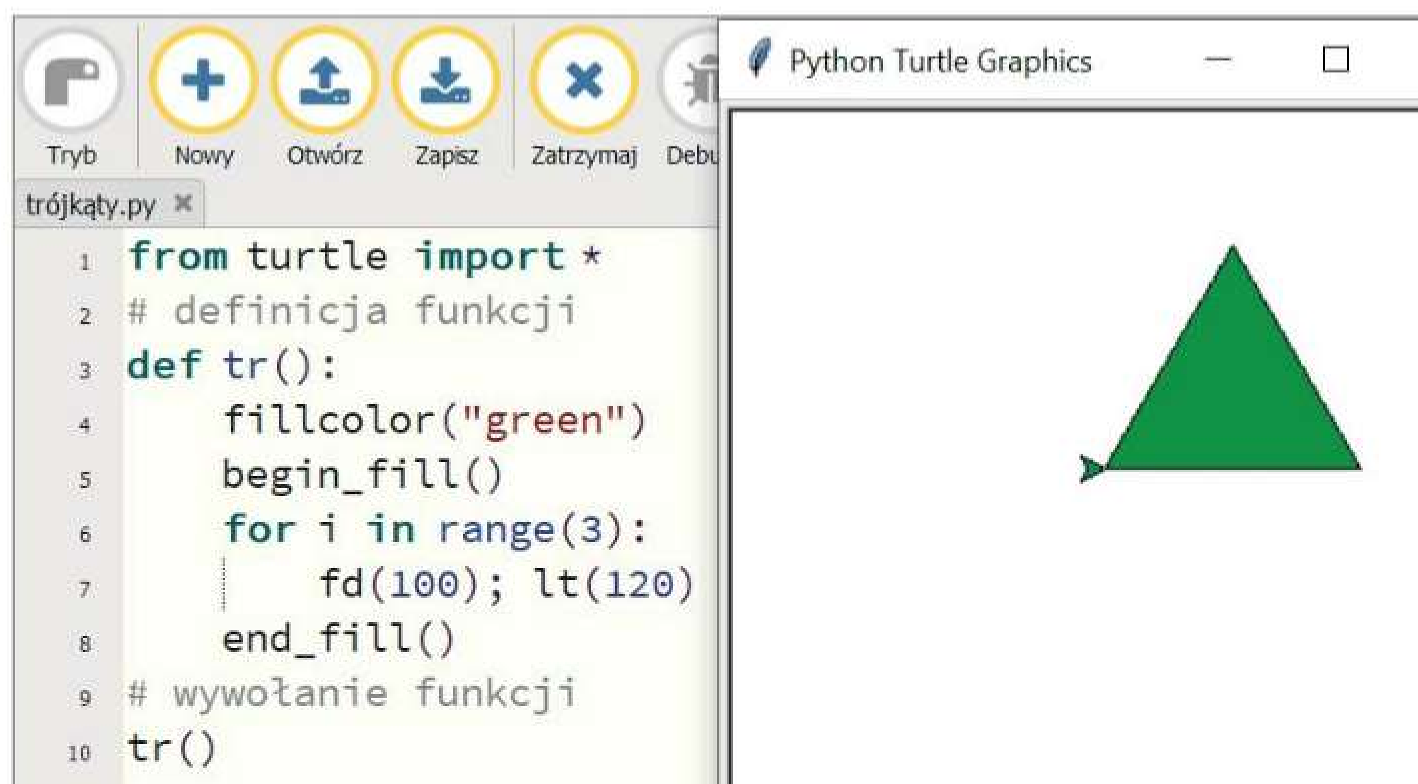
- Zaimportuj moduł **`turtle`**.
- Napisz słowo kluczowe **`def`** i nazwę funkcji, następnie wstaw pusty nawias oraz dwukropek i naciśnij klawisz **Enter**, aby Python zrobił odpowiednie wcięcie.
- Zapisz polecenia do wykonania w funkcji – w przypadku rysowania kwadratu jest to pętla **`for`**, która czterokrotnie powtarza rysowanie boku długości 100 i obrót o kąt 90°.
- W kolejnym wierszu, już bez wcięcia, wpisz wywołanie funkcji, tj. **`kwadrat()`**.



Rys. 1. Definicja i wywołanie funkcji oraz efekt

A teraz zdefiniuj funkcję **`tr()`** rysowania zamalowanego na zielono trójkąta równobocznego o boku 100.

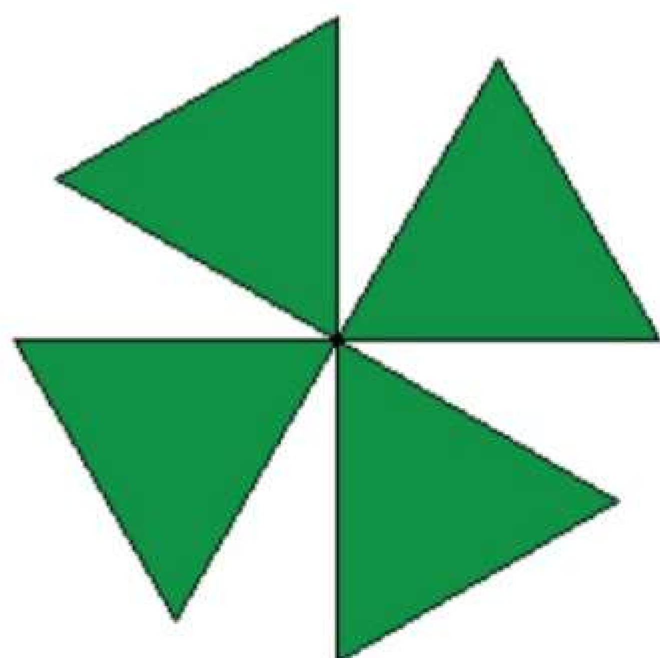
- Zaimportuj moduł **`turtle`**.
- Napisz słowo kluczowe **`def`** i nazwę funkcji, wstaw pusty nawias oraz dwukropek i naciśnij **Enter**.
- Zapisz polecenia do wykonania w funkcji – upewnij się, że wcięcia wyróżniające instrukcje o tej samej wadze (wiersze 4, 5, 6 i 8) są jednakowej wielkości.
- Zapisz polecenie wywołania funkcji.



Rys. 2. Definicja i wywołanie funkcji oraz efekt

Funkcja może również zawierać wywołanie innej funkcji.

Zdefiniuj funkcję **`wiatrak()`** odpowiadającą za rysowanie wiatraka złożonego z czterech trójkątów. Wykorzystaj funkcję **`tr()`** rysowania trójkąta.



Rys. 3. Wiatrak z czterech trójkątów

- Przeanalizuj problem: rysowanie wiatraka polega na narysowaniu czterech trójkątów, przy czym każdy kolejny trójkąt jest ustawiony pod kątem prostym do poprzedniego; w definicji warto wykorzystać pętlę **`for`**.

- Dopisz definicję funkcji **wiatrak()** pod definicją funkcji rysowania trójkąta.
  - Zostaw wiersz odstępu między funkcjami, aby uzyskać lepszą czytelność kodu.
  - Zacznij pisać bezpośrednio przy lewym marginesie.
- Usuń zbędne wywołanie funkcji i dodaj właściwe.

```
from turtle import *
# definicja funkcji
def tr():
    fillcolor("green")
    begin_fill()
    for i in range(3):
        fd(100); lt(120)
    end_fill()

def wiatrak():
    for i in range(4):
        tr(); rt(90)
# wywołanie funkcji
wiatrak()
```

Rys. 4. Przykładowy skrypt rysowania wiatraka

DEFINIOWANIE FUNKCJI Z PARAMETREM

Jak się zapewne domyślasz, w przypadku funkcji bez parametru każda zmiana wartości pociąga za sobą edycję kodu. W przypadku długości boku kwadratu czy trójkąta korektę wprowadzasz w jednym miejscu, ale bardziej skomplikowane programy mogą wymagać znacznych modyfikacji. Dlatego do nawiasu po nazwie funkcji warto wstawić nazwę parametru, a jego konkretną wartość podać podczas wywołania funkcji.

Zdefiniuj funkcję **kwadrat(bok)** odpowiadającą za rysowanie kwadratu o podanym boku.

- Zaimportuj moduł **turtle**.
- Napisz słowo kluczowe **def** i nazwę funkcji, następnie wstaw nawias i umieść w nim nazwę parametru, tj. **bok**. Dodaj dwukropek.
- Zapisz pętlę **for**, która czterokrotnie powtarza rysowanie boku o pewnej długości (tj. o wartości parametru **bok**) i obrót o kąt 90°.
- Sprawdź działanie funkcji dla kilku wartości parametrów, np. 60, 110 i 150. W wywołaniu funkcji wpisz do nawiasu wybraną wartość.
  - Każde wywołanie sprawdź w osobnej sesji (**Wykonuj** → **Zatrzymaj**).
  - Czy domyślasz się, co się stanie, gdy w skrypcie umieścisz wszystkie trzy wywołania naraz?

```
from turtle import *
# definicja funkcji
def kwadrat(bok):
    for i in range(4):
        fd(bok); lt(90)
# wywołanie funkcji
kwadrat(60)
```

Rys. 5. Definicja i wywołanie funkcji **`kwadrat`** z parametrem **`bok`**

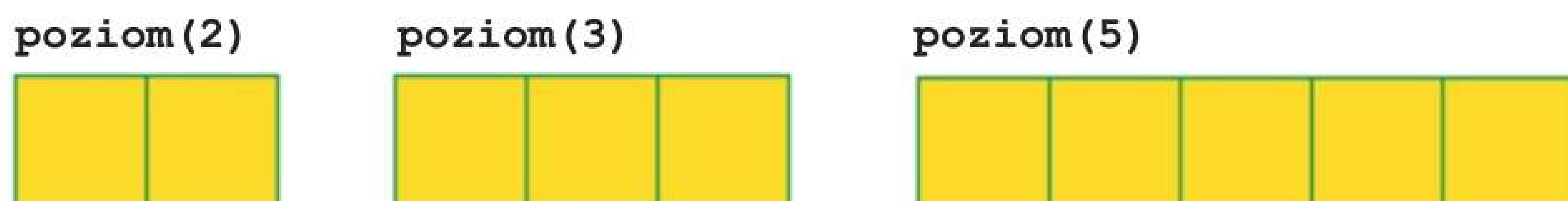
Dopisz fragment kodu tak, aby kwadrat był narysowany kolorem zielonym, a zamalowany kolorem złotym.

- Uzupełnij odpowiednio definicję funkcji – polecenia **`pencolor()`**, **`fillcolor()`** i **`begin_fill()`** powinny znaleźć się przed pętlą **`for`**, natomiast polecenie **`end_fill()`** powinno zakończyć działanie całej funkcji.
  - Zwróć uwagę na jednakowe wcięcia dla poleceń tej samej wagi.



Rys. 6. Definicja i wywołanie nowej funkcji **`kwadrat`**

A teraz zdefiniuj funkcję **`poziom`** z parametrem **`ile`** odpowiadającą za rysowanie podanej liczby kwadratów o boku 40, ustawionych jeden za drugim w poziomie. Wykorzystaj funkcję **`kwadrat(bok)`** rysowania jednego kwadratu.



Rys. 7. Efekty wywołania funkcji **`poziom`** dla parametru o wartości **2**, **3** i **5**

- Przeanalizuj problem: po narysowaniu jednego kwadratu o boku 40 żółw musi się przemieścić o wielkość równą długości boku, aby poprawnie narysować kolejny kwadrat, co oznacza, że w skrypcie należy wykorzystać pętlę **`for`**.

- Zaimportuj moduł **`turtle`**.

- Zapisz pierwszy wiersz definicji funkcji. Pamiętaj o dwukropku.

- Zapisz pętlę **for**.
  - Zauważ, że liczba obrotów pętli powinna być równa liczbie kwadratów, co oznacza, że parametrem funkcji **range()** jest parametr **ile**, który informuje o liczbie kwadratów.
  - Polecenia wykonywane w pętli to rysowanie kwadratu o boku 40 (a więc wywołanie funkcji **kwadrat(40)** i przesunięcie żółwia o 40 kroków do przodu).

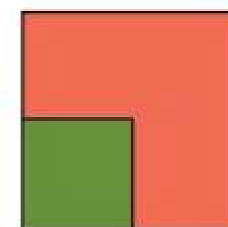
```
def poziom(ile):
    for i in range(ile):
        kwadrat(40); fd(40)
# wywołanie funkcji
poziom(3)
```

Rys. 8. Definicja i wywołanie funkcji **poziom** z parametrem **ile**

POSADZKI

Teraz możesz się zająć rysowaniem kwadratowych posadzek, pokrytych przylegającymi i niezachodzącymi na siebie kwadratowymi kafelkami.

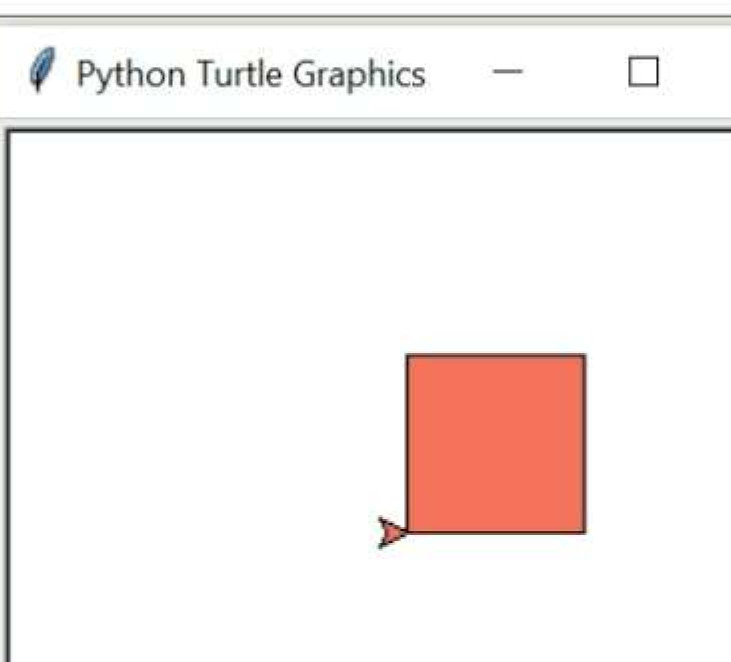
Zacznij od narysowania pojedynczego kafelka, czyli zdefiniowania funkcji **kafelek** z parametrem **bok**. Można to zrobić na różne sposoby – liczy się efekt w postaci rysunku. Jednym ze sposobów jest nakładanie figur na siebie. W przypadku kafelka przedstawionego na rys. 9 na czerwony kwadrat można nałożyć kwadrat oliwkowy o boku o połowę mniejszym – wystarczy narysować oba kwadraty jeden po drugim w odpowiedniej kolejności.



Rys. 9. Wzorcowy kafelek

- Zaimportuj moduł **turtle**.
- Zdefiniuj funkcję **kwadrat** z parametrami **bok** i **kolor**, która będzie rysowała kwadrat o podanym boku i zamalowany podanym kolorem.
- Sprawdź działanie funkcji dla różnych wartości parametru **kolor**.

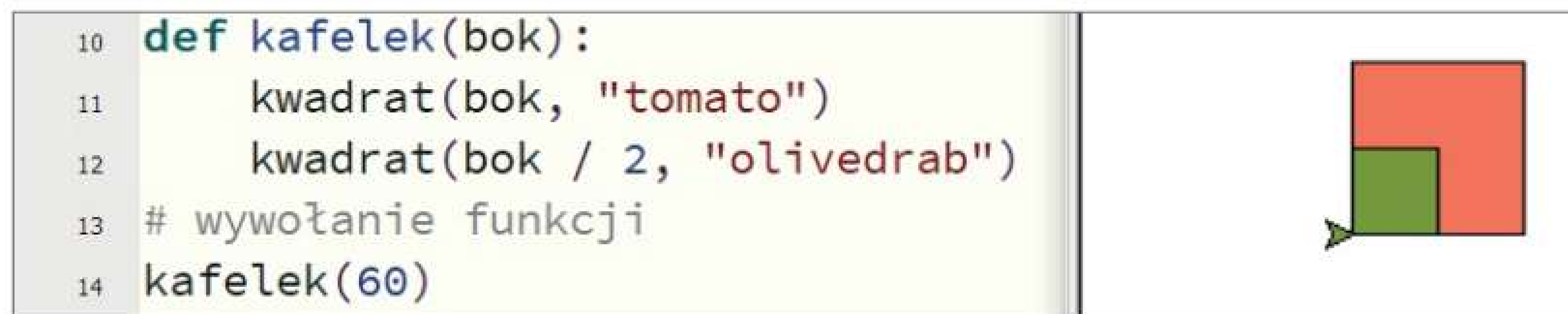
```
from turtle import *
# definicja funkcji
def kwadrat(bok, kolor):
    fillcolor(kolor)
    begin_fill()
    for i in range(4):
        fd(bok); lt(90)
    end_fill()
# wywołanie funkcji
kwadrat(60, "tomato")
```



Rys. 10. Rysowanie pierwszego kwadratu

- Jeśli zamalowane kwadraty rysują się prawidłowo, skasuj wywołanie funkcji. Zostaw wiersz odstępu.

- Zdefiniuj funkcję **kafelek(bok)** i przetestuj jej działanie.
  - Wykorzystaj funkcję **kwadrat** do narysowania dwóch kwadratów – zwróć uwagę na sposób deklaracji boku mniejszego kwadratu oraz opis parametru **kolor**.

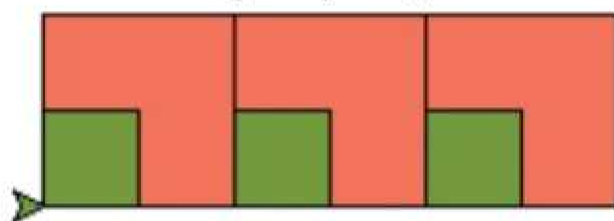


```
def kafelek(bok):
    kwadrat(bok, "tomato")
    kwadrat(bok / 2, "olivedrab")
# wywołanie funkcji
kafelek(60)
```

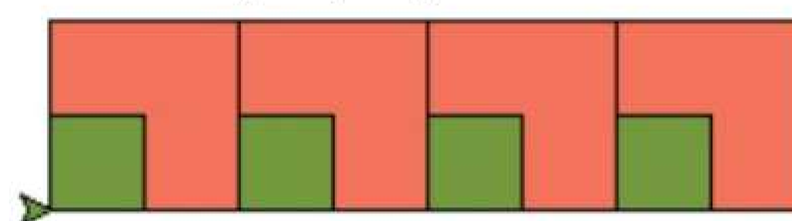
Rys. 11. Rysowanie kafelka

Teraz możesz rozszerzyć program i zdefiniować funkcję **warstwa** z parametrami **bok** i **ile**, odpowiadającą za rysowanie danej liczby kafelków ułożonych w poziomie jeden obok drugiego, tak jak na rysunku poniżej. Po zakończeniu rysowania żółw powinien wrócić do pozycji początkowej (takie przenoszenie żółwia do punktu startu jest częstą praktyką, gdyż ułatwia rysowanie kolejnych elementów).

warstwa(60, 3)

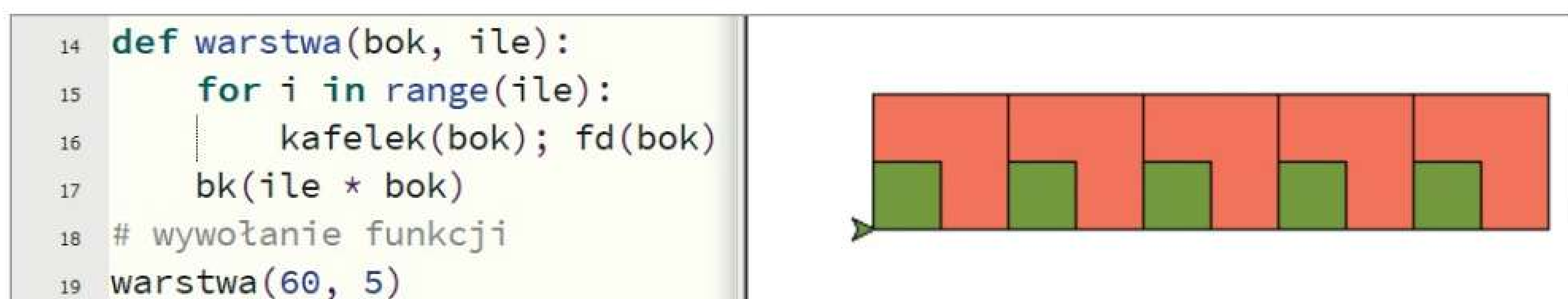


warstwa(60, 4)



Rys. 12. Warstwy wyrysowane dla parametrów **(60, 3)** i **(60, 4)**

- Uzupełnij skrypt – zdefiniuj funkcję **warstwa(bok, ile)** i przetestuj jej działanie.
  - Zastanów się, ile razy powinna zostać wykonana pętla, i odpowiednio uzupełnij parametr funkcji **range()**.
  - Zauważ, że powrót do pozycji początkowej nie jest elementem pętli, dlatego należy odpowiednio zmniejszyć wcięcie.

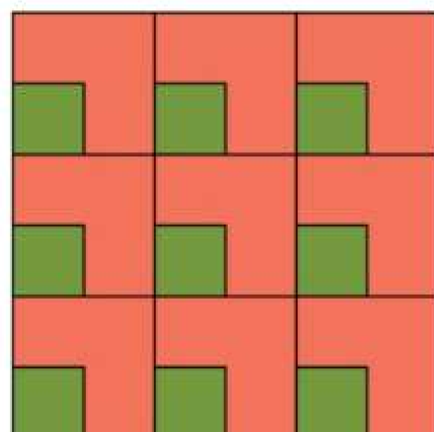


```
def warstwa(bok, ile):
    for i in range(ile):
        kafelek(bok); fd(bok)
    bk(ile * bok)
# wywołanie funkcji
warstwa(60, 5)
```

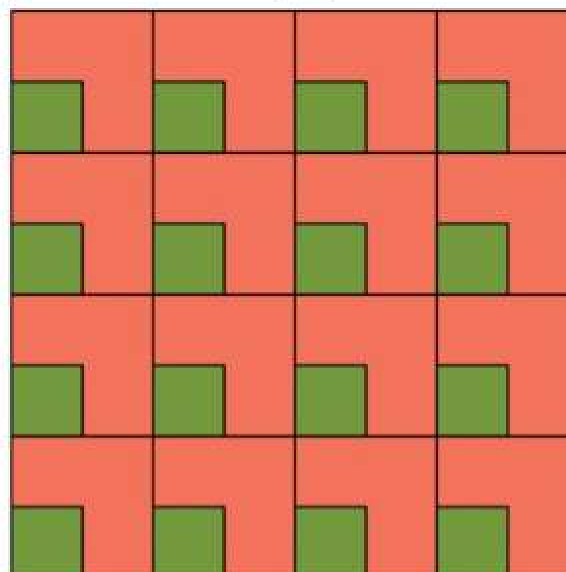
Rys. 13. Rysowanie warstwy pięciu kafelków

Pozostało napisanie funkcji **posadzka** z parametrem **ile**, po wywołaniu której zostanie narysowana kwadratowa posadzka składająca się z **ile** × **ile** kafelków o długości boku 60.

posadzka(3)



posadzka(4)



Rys. 14. Posadzki wyrysowane dla parametrów 3 i 4

► Uzupełnij skrypt – zdefiniuj funkcję **posadzka(ile)** i przetestuj jej działanie.

- Ponieważ długość boku kafelka jest stała i wynosi 60, to aby ta wartość została zapamiętana, skorzystaj ze zmiennej, np. **bok**. Do jej wartości możesz odwoływać się w całym programie, podobnie jak do parametru.
- Zauważ, że tym razem pętla powinna zostać wykonana **ile** razy – uzupełnij odpowiednio parametr funkcji **range()**.
- Żółw w pętli powinien narysować odpowiednią liczbę kafelków ułożonych w poziomie jeden obok drugiego, a następnie przemieścić się tak, aby narysować kolejną warstwę.

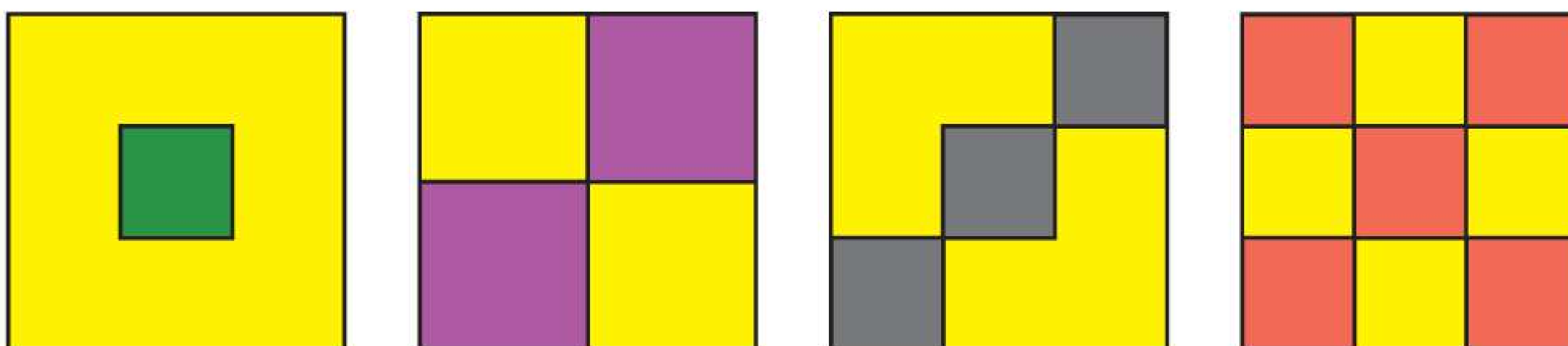
```

def posadzka(ile):
    bok = 60
    for i in range(ile):
        warstwa(bok, ile)
        pu(); rt(90); fd(bok); lt(90); pd()
# wywołanie funkcji
posadzka(4)

```

Rys. 15. Rysowanie posadzki

Napisz funkcje rysowania wybranego kafelka. Pamiętaj o tym, aby żółw zaczynał w lewym dolnym rogu i w tym samym miejscu kończył rysowanie, wtedy funkcje **wiersz** i **posadzka** będą prawidłowo działały również dla nowego kafelka.

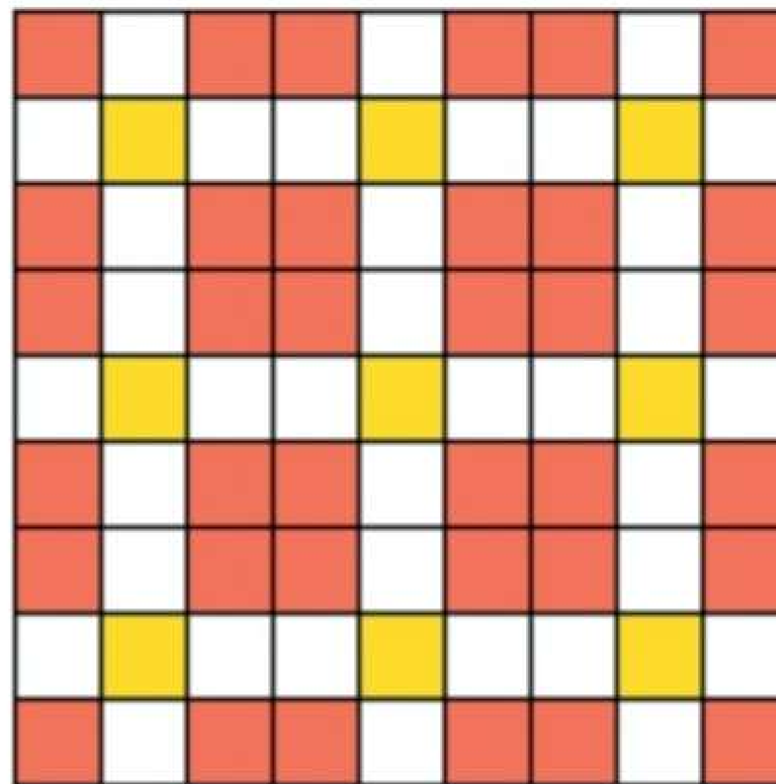


Rys. 16. Przykłady kwadratowych kafelków

ZADANIA

1. Zaprojektuj swój wzór i kolorystykę kafelka, a następnie napisz funkcję rysowania takiego kafelka.

2. Napisz funkcję **`dywan(ile)`** odpowiedzialną za rysowanie takiego kwadratowego dywanu jak na rysunku poniżej (efekt po wywołaniu funkcji dla parametru 3).



2.3 Pisz i powtarzaj

DOWIESZ SIĘ, JAK

- wypisywać na ekranie tekst,
- prowadzić dialog z użytkownikiem.

Po zabawach z żółwiem możesz przystąpić do pisania pierwszego tekstowego programu w Pythonie. Choć rysunków i teraz nie zabraknie, będziesz rysować na ekranie w inny sposób – za pomocą polecenia `print`.

PRINT, CZYLI WYPISZ

Polecenie `print` wyświetla na ekranie tekst zapisany jako jego parametr w nawiasie i cudzysłowie lub apostrofach. W edytorze Mu efekt uruchomienia programu widać w oknie **Wykonywanie** na dole programu.

```
print("Mam na imię Ania.")
```

Wykonywanie: 1.py

```
Mam na imię Ania.
>>>
```

Rys. 1. Polecenie `print` i jego efekt

Może się zdarzyć, że zamiast napisanego programu wyświetli się **SyntaxError** (czytaj: syntakserer), czyli błąd składni. Wówczas należy poprawić błąd i ponownie uruchomić program.

```
print("Mam na imię Ania.)
```

Wykonywanie: 1.py

```
File "c:\03\1.py", line 1
  print("Mam na imię Ania.)
                          ^
SyntaxError: EOL while scanning string literal
>>>
```

Uwaga dotyczy pierwszego wiersza w pliku o nazwie **1.py**.

Podczas skanowania ciągu znaków wystąpił błąd składni na końcu wiersza (EOL – End of Line).

Rys. 2. Przykładowy błąd składni – przed nawiasem zamykającym brakuje znaku "

▶ Dopisz drugie polecenie **print**, a następnie uruchom program.

```
print("Mam na imię Ania.")
print("Uczę się programować w Pythonie.")

Wykonywanie: 1.py

Mam na imię Ania.
Uczę się programować w Pythonie.
>>>
```

Rys. 3. Dwa polecenia **print** i ich efekt

Narysuj trójkąt z wykorzystaniem znaku *****. Wykorzystaj polecenie **print**.

```
*********
 *******
  *****
   ***
    *
```

Rys. 4. Trójkąt z gwiazdek

▶ Przeanalizuj problem:

- podstawę trójkąta tworzy dziewięć gwiazdek;
- w każdym kolejnym wierszu są o dwie gwiazdki mniej, za to pojawiają się spacje – w drugim wierszu jest jedna spacja z lewej strony, w trzecim – dwie spacje, w czwartym – trzy spacje, w ostatnim – cztery;
- oznacza to, że w pierwszym wierszu trzeba wypisać dziewięć gwiazdek, w drugim – jedną spację i siedem gwiazdek itd.

▶ Zapisz kolejne instrukcje – pamiętaj o domknięciu nawiasów i cudzysłowów.

```
print("*********")
print(" *******")
print("  *****")
print("   ***")
print("    *")
```

Rys. 5. Instrukcje rysowania trójkąta

Zmodyfikuj kod tak, aby wypisywał trójkąt z rys. 4, ale stojący na podstawie.

```
    *
   ***
  *****
 *******
*********
```

Rys. 6. Trójkąt z gwiazdek wersja 2

- Przeanalizuj problem: co trzeba zrobić, aby na górze była jedna gwiazdka, a na dole dziewięć gwiazdek?

- Zmień kolejność wypisywania gwiazdek i sprawdź działanie programu.

```
1 print("    *")
2 print("   ***")
3 print("  *****")
4 print(" *******")
5 print("*********")
```

Rys. 7. Wypisywanie trójkąta z gwiazdek

A jak narysować pusty trójkąt, który ma tylko krawędź z gwiazdek?

- Przeanalizuj problem: trzeba narysować pełny trójkąt, a następnie usunąć gwiazdki, które znajdują się w jego wnętrzu, i wstawić w ich miejsce spacje.

- Skopiuj kod wypisywania poprzedniego trójkąta, zastąp odpowiednie gwiazdki spacjami i sprawdź, czy trójkąt rysuje się poprawnie.

```
1 print("    *")
2 print("   * *")
3 print("  *   *")
4 print(" *     *")
5 print("*********")
```

→

```
    *
   * *
  *   *
 *     *
*********
```

Rys. 8. Wypisywanie pustego trójkąta z gwiazdek

Rozbuduj program rysowania poprzedniego trójkąta tak, aby powstała taka strzałka jak na rysunku poniżej.

```
    *
   * *
  *   *
 *     *
***   ***
  *   *
  *****
```

Rys. 9. Strzałka z gwiazdek

- Przeanalizuj problem: musisz zmodyfikować ostatni wiersz wypisywania gwiazdek (usunąć trzy środkowe gwiazdki z podstawy) oraz dopisać dwa dodatkowe wiersze.

- Powiel kod wypisywania trójkąta, a następnie zmodyfikuj odpowiednio wiersz 5, skopiuj wiersz 3, dopisz wiersz 7 i sprawdź, czy strzałka rysuje się poprawnie.

```
print("    *")
print("   * *")
print("  *   *")
print(" *     *")
print("***   ***")
print("  *   *")
print("  *****")
```

Rys. 10. Wypisywanie strzałki z gwiazdek

Na koniec napisz program wypisywania dwóch choinek, które stoją obok siebie. Wykonaj rysunek według własnego pomysłu – może wyglądać tak jak poniżej.

```
print("    *        *")
print("   ***      ***")
print("  *****    *****")
print(" *******  *******")
print("*****************")
print("   ***      ***")
```

```
Wykonywanie: 2choinki.py
    *        *
   ***      ***
  *****    *****
 *******  *******
*****************
   ***      ***
```

Rys. 11. Wypisywanie dwóch choinek z gwiazdek

WCZYTYWANIE DANYCH

Wczytywanie danych realizuje się za pomocą funkcji **`input()`**, której parametrem może być tekst. Po wywołaniu funkcji program czeka na odpowiedź użytkownika. W edytorze Mu należy ją wpisać w oknie **Wykonywanie** i zatwierdzić klawiszem **Enter**.

► Przepisz do edytora poniższy program. Zwróć uwagę na spację przed cudzysłowem w wierszu 1 (to miejsce na wpisanie odpowiedzi przez użytkownika) oraz przecinek przed zmienną **`odp`** w wierszu 2.

```
1 odp = input("Jak masz na imię? ")
2 print("Cześć,",odp,"!")
3
```

```
Wykonywanie: dialog0.py
Jak masz na imię? Ania
Cześć, Ania !
>>>
```

Odpowiedź wpisana przez użytkownika.

Rys. 12. Dialog komputera z użytkownikiem

► Przeanalizuj zastosowany mechanizm. Po naciśnięciu przycisku **Wykonaj** następuje wywołanie funkcji **input()**, program czeka na odpowiedź użytkownika, a tekst wpisany z klawiatury zostaje zapamiętany jako wartość zmiennej **odp**. Polecenie **print** powoduje wypisanie na ekranie zadeklarowanego w nawiasie napisu oraz wartości podanej zmiennej.

A teraz rozbuduj dialog z użytkownikiem. Zapytaj go o ulubiony kolor.

► Zdefiniuj zmienną przechowującą tekst wpisany przez użytkownika i wprowadź instrukcję wypisywania tekstu zawierającego odpowiedź użytkownika.
- Zadbaj o to, by odpowiedź wpisywana przez użytkownika nie sklejała się z zadawanym pytaniem.
- Podczas przygotowywania odpowiedzi komputera pamiętaj o domknięciu cudzysłowu i dodaniu przecinka przed zmienną.

```
odp = input("Jak masz na imię? ")
print("Cześć,",odp,"!")
odp = input("Mój ulubiony kolor to zielony, a Twój? ")
print(odp, "to też ładny kolor.")
```

Wykonywanie: dialog.py

```
Jak masz na imię? Ania
Cześć, Ania !
Mój ulubiony kolor to zielony, a Twój? Niebieski
Niebieski to też ładny kolor.
>>>
```

Rys. 13. Dialog z użytkownikiem

ZADANIA

1. Napisz trzy programy wypisujące znaki @ według poniżej pokazanych wzorów.

```
@@
@@
@@
@@
@@@@@@
@@@@@@
```

```
@@@@@@
  @@
  @@
  @@
  @@
```

```
@@@@@@@
 @@@@@
  @@@
   @
  @@@
 @@@@@
@@@@@@@
```

2. Ułóż krótki dialog komputera z użytkownikiem na temat ostatnio przeczytanej książki lub ulubionej gry komputerowej.

2.4 Proste obliczenia

DOWIESZ SIĘ, JAK

- posługiwać się podstawowymi operatorami arytmetycznymi w Pythonie,
- pisać proste programy wykonujące obliczenia,
- wypisywać wyniki.

Python pozwala na wykonywanie obliczeń. Podczas tej lekcji nauczysz się odpowiednio zapisywać liczby i działania oraz tworzyć proste kalkulatory realizujące przeliczanie jednostek.

OPERACJE ARYTMETYCZNE

W języku Python możesz dodawać (`+`), odejmować (`–`), mnożyć (`*`), dzielić (`/` lub `//`), wyznaczać resztę z dzielenia (`%`) i potęgować (`**`).

► Otwórz edytor Mu, wpisz polecenia z rys. 1, uruchom program i sprawdź otrzymane wyniki. Zwróć uwagę na zapis liczby w postaci dziesiętnej w czwartym wierszu – przecinek zastąpiony jest kropką.

Tryb | Nowy | Otwórz | Zapisz | Zatrzymaj | Debuguj | REPL | Plotter

operacje.py

```
1 print(5 + 2) # dodawanie
2 print(5 - 2) # odejmowanie
3 print(5 * 2) # mnożenie
4 print(5 / 2) # dzielenie
5 print(5 // 2) # dzielenie całkowite
6 print(5 % 2) # reszta z dzielenia całkowitego
7 print(5 ** 2) # potęgowanie
```

Wykonywanie: operacje.py

```
7
3
10
2.5
2
1
25
>>>
```

Rys. 1. Operacje arytmetyczne w języku Python

ZMIENNE

Jak wiesz, w Pythonie można stosować zmienne, które służą do przechowywania danych w pamięci komputera (np. zmienna **`odp`** w lekcji 2.3). Program udostępnia kilka wbudowanych typów zmiennych, np. liczby całkowite (typu **`int`**) i rzeczywiste (typu **`float`**) oraz napisy (**`string`**).

Podczas tworzenia zmiennej nie określa się jej typu, tylko podaje nazwę i wartość. Należy przy tym pamiętać, że:

- nazwa zmiennej nie może rozpoczynać się od cyfry ani zawierać polskich znaków, ale może być poprzedzona podkreślnikiem (poprawnie zapisane nazwy zmiennych to np. **`n`**, **`_droga`**, **`bok1`**);
- wartość nadaje się (przypisuje) za pomocą pojedynczego znaku równości =;
- wielkość liter w nazwie ma znaczenie; **`A = 40`** i **`a = 40`** to różne zmienne, mające te same wartości.

▶ Utwórz i wypisz kolejno zmienną typu całkowitego, rzeczywistego i napisowego. Kieruj się opisanymi wyżej zasadami.

```
a = 12
print(a)
x1 = 145.4
print(x1)
s = "wakacje"
print(s)
_s = "koniec wakacji"
print(_s)
```

Wykonywanie: zmienne.py

```
12
145.4
wakacje
koniec wakacji
>>>
```

liczba całkowita
liczba rzeczywista
napis
napis

Rys. 2. Przykładowe zmienne i ich wartości

▶ Poeksperymentuj.

- Sprawdź, co się stanie, jeśli dodasz liczby, np. **`print(x1 + x1)`** lub napisy, np. **`print(s + s)`**.
- Wstaw symbol **`" "`** jako jeden ze składników dodawania napisów, aby dodane napisy były oddzielone spacją, np. **`print(s + " " + s)`**.

OBLICZENIA

Czas zająć się obliczeniami. Na początek coś niezbyt skomplikowanego. Aby stworzyć prosty kalkulator, który będzie wyznaczać pole kwadratu o zadanym boku, należy zadeklarować zmienną, np. **`bok = 5`**, a następnie wypisać iloczyn **`bok * bok`** lub potęgę **`bok ** 2`**.

```
bok = 5
print(bok * bok)
print(bok ** 2)
```

Wykonywanie: pole.py

```
25
25
>>>
```

Rys. 3. Wyznaczanie pola kwadratu o boku równym 5

Aby stworzyć konwerter jednostek prędkości, za pomocą którego wartość prędkości podaną w kilometrach na godzinę można wyrazić w metrach na sekundę, należy zadeklarować zmienną, np. **`v = 180`**, a następnie wypisać iloraz **`v / 3.6`** (1 km = 1000 m, 1 h = 3600 s, 1 km/h = 1000 m/3600 s = 1 m/3,6 s, wobec tego trzeba podzielić wartość prędkości przez 3,6).

```
v = 180
print(v / 3.6)
```

Wykonywanie: prędkość.py

```
50.0
>>>
```

Rys. 4. Przeliczanie prędkości 180 km/h na m/s

Jak widać, w takim programie za każdym razem trzeba zmieniać wartość przypisaną zmiennej w skrypcie. Jak stworzyć program niewymagający wprowadzania modyfikacji w kodzie? Można wykorzystać poznane na poprzedniej lekcji wczytywanie danych.

Napisz program, w którym temperatura (liczba całkowita) podana przez użytkownika w stopniach Fahrenheita zostanie przeliczona na stopnie Celsjusza (w razie potrzeby wpisz w wyszukiwarkę hasło „Fahrenheit na Celsjusz” i odszukaj wzór umożliwiający przeliczenie wartości temperatury).

- Zadeklaruj zmienną przechowującą wartość wczytanej liczby całkowitej podanej przez użytkownika, np. **`f`**. Zastosuj funkcję **`int()`**, dzięki której użytkownik zostanie poinformowany o błędzie, jeżeli poda liczbę niecałkowitą.
- Zadeklaruj zmienną, której wartość będzie stanowił wynik przeliczenia jednostek, np. **`c`** – przy zapisywaniu wzoru zastosuj odpowiednie dzielenie i właściwy zapis liczby w postaci dziesiętnej.
- Wypisz wynik przechowywany przez zmienną **`c`**.

```
f = int(input("Podaj temperaturę w stopniach Fahrenheita: "))
c = (f - 32) / 1.8
print(c)
```

Rys. 5. Przeliczanie stopni Fahrenheita na stopnie Celsjusza

ZADANIA

1. Napisz program, w którym użytkownik podaje liczbę całkowitą – długość w centymetrach, a w wyniku otrzymuje wartość długości przeliczoną na metry i cale.

2. Węzeł to jednostka stosowana do podawania prędkości statków morskich i powietrznych, równa jednej mili morskiej na godzinę. Samolot pokonał trasę z Nowego Jorku do Warszawy z prędkością 528 węzłów. Napisz program, który obliczy i wypisze prędkość tego samolotu wyrażoną w kilometrach na godzinę.



2.5 Pętle i warunki

DOWIESZ SIĘ, JAK

- zmieniać wartość początkową zmiennej,
- stosować pętlę **`while`**,
- stosować instrukcję warunkową.

Podczas tej lekcji utrwalisz i poszerzysz umiejętności zdobyte na poprzednich lekcjach, dzięki czemu będziesz bez problemów wykonywać nawet bardziej złożone obliczenia w Pythonie.

ZMIANA WARTOŚCI ZMIENNEJ

Podczas programowania może zajść potrzeba zmiany wartości zmiennej – np. możesz chcieć zwiększyć poprzednią wartość zmiennej **`temp`** o jakąś liczbę. Masz do wyboru dwa sposoby zapisu przedstawione na rys. 1. Oba powodują dodanie liczby do poprzedniej wartości zmiennej. Drugi zapis jest skróconą formą pierwszego.

► Przeanalizuj poniższy skrypt, a następnie sprawdź, jak działają te operacje w przypadku odejmowania (**`-=`**), mnożenia (**`*=`**) i dzielenia (**`/=`**).

Tryb Nowy Otwórz Zapisz Zatrzymaj Debuguj REPL

temp.py

```
# wartość początkowa zmiennej
temp = 1
print(temp)
# zwiększenie wartości zmiennej o 4
temp = temp + 4
print(temp)
# zwiększenie wartości zmiennej o 7
temp += 7
print(temp)
```

Wykonywanie: temp.py

```
1
5
12
>>>
```

Rys. 1. Dwa sposoby zapisu zwiększania wartości zmiennej **`temp`**

SUMA CYFR

Skorzystaj z informacji na temat zmiany wartości zmiennej i napisz program, który obliczy i wypisze sumę cyfr czterocyfrowej liczby całkowitej dodatniej.

- Przeanalizuj problem. Jeżeli masz daną liczbę $n = 1048$, to w wyniku otrzymasz $1 + 0 + 4 + 8 = 13$. Aby wyodrębnić poszczególne cyfry, możesz wykorzystać dwie operacje: modulo 10, która wyodrębni ostatnią cyfrę, oraz dzielenie całkowite przez 10, za pomocą którego pozbędziesz się ostatniej cyfry. Obie operacje musisz powtórzyć czterokrotnie – dla każdej cyfry:
  1048 mod 10 = **8** → 1048 // 10 = 104
  104 mod 10 = **4** → 104 // 10 = 10
  10 mod 10 = **0** → 10 // 10 = 1
  1 mod 10 = **1** → 1 // 10 = 0
- Zadeklaruj zmienną, która ma przechowywać przetwarzaną liczbę.
- Zadeklaruj zmienną, która ma przechowywać wartość sumy, i nadaj tej zmiennej wartość początkową równą 0. Będziesz do niej dodawać kolejne wyodrębnione cyfry.
- Zapisz pętlę `for` – zastosuj algorytm, zgodnie z którym wyodrębniona z danej liczby ostatnia cyfra zostanie dodana do sumy, a następnie usunięta.
- Wypisz wyznaczoną sumę.
- Uruchom program dla kilku wartości. Sprawdź poprawność obliczeń.

```python
liczba = 4832
suma = 0
for i in range(4):
    # ostatnia cyfra
    suma = suma + liczba % 10
    # liczba bez ostatniej cyfry
    liczba = liczba // 10
print(suma)
```

```
17
>>>
```

Rys. 2. Przykładowy program, który oblicza sumę cyfr liczby czterocyfrowej

Rozwiązanie tego zadania można uogólnić dla liczby o dowolnej długości.

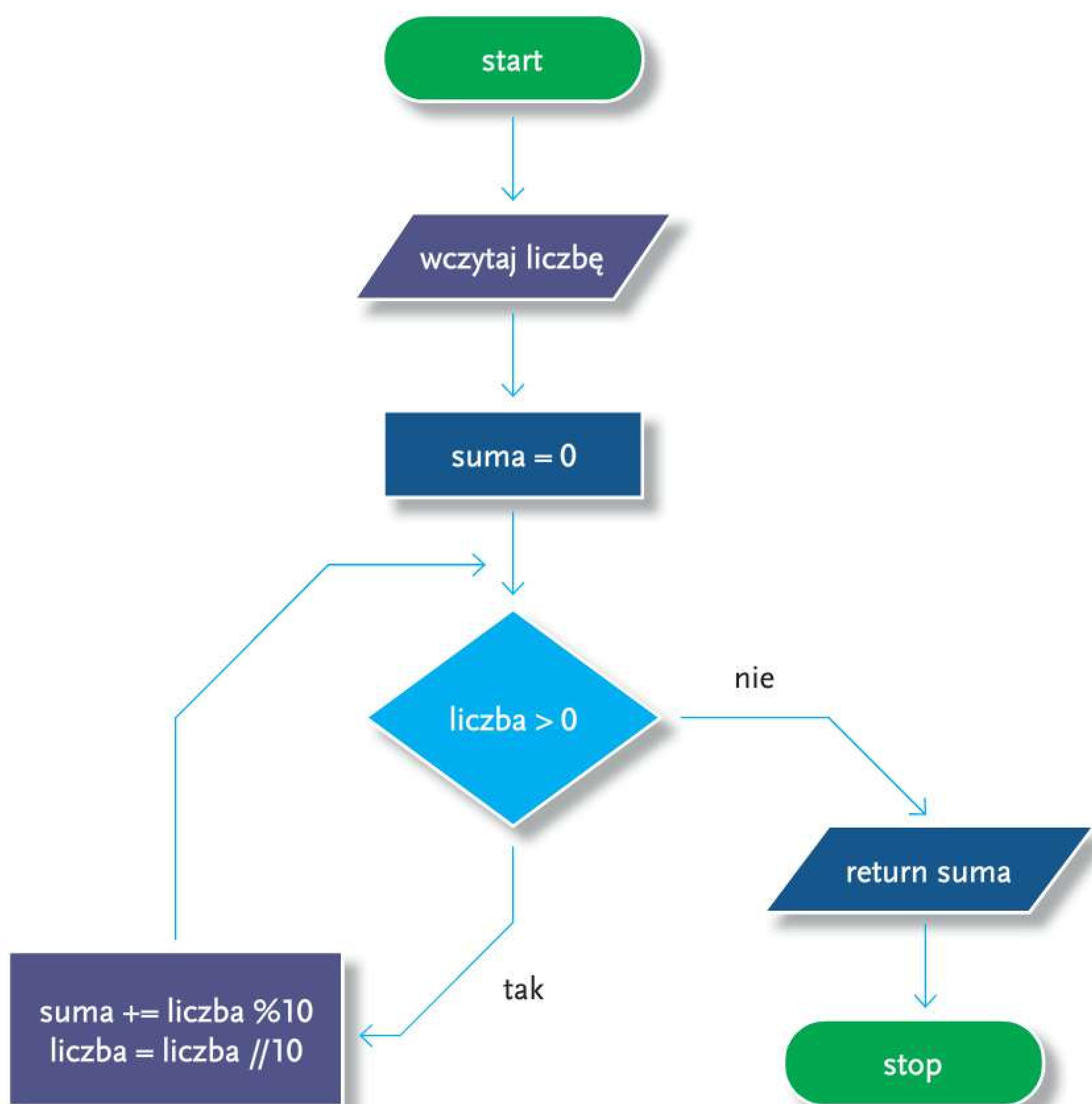
PĘTLA WHILE

Jeśli nie wiesz, ile cyfr ma liczba, najlepiej zastosować pętlę **`while`**. Wykonuje ona ciąg poleceń, dopóki podany warunek jest prawdziwy.

```
while <warunek>:
    <ciągi poleceń>
```

Rys. 3. Składnia pętli **`while`**

Jak to wygląda w przypadku obliczania i wypisywania sumy cyfr liczby całkowitej dodatniej? Pokazuje to poniższy schemat blokowy.



Rys. 4. Schemat blokowy algorytmu obliczania sumy cyfr dowolnej liczby

Najpierw trzeba wczytać liczbę. Niech będzie równa 483 284. Zmiennej **`suma`** należy przypisać wartość 0. Następnie należy sprawdzić, czy dana liczba jest większa od 0. Jeśli nie, to program powinien wypisać na ekranie aktualną sumę i zakończyć działanie. Jeśli tak – a tak jest w tym przypadku – to należy wykonać kolejne obliczenia:

- **`suma = suma + liczba % 10`**, czyli **`suma = 0 + 483284 % 10 = 0 + 4 = 4`**;
- **`liczba = liczba // 10`**, czyli **`liczba = 483284 // 10 = 48328`**.

Na koniec należy sprawdzić, czy liczba jest większa od 0. Tak jest, a zatem:

- **`suma = suma + liczba % 10`**, czyli **`suma = 4 + 48328 % 10 = 4 + 8 = 12`**;
- **`liczba = liczba // 10`**, czyli **`liczba = 48328 // 10 = 4832`**.

Czy liczba jest większa od 0?

...

- **`suma = suma + liczba % 10`**, czyli **`suma = 25 + 4 % 10 = 25 + 4 = 29`**;
- **`liczba = liczba // 10`**, czyli **`liczba = 4 // 10 = 0`**.

Czy liczba jest większa od 0?
Nie.
Suma cyfr liczby 483284 została wyznaczona i wynosi 29.

Powyższy algorytm zapisany w Pythonie może wyglądać tak jak na rys. 5.

Tryb Nowy Otwórz Zapisz Zatrzymaj Debuguj REPL Plotter Powiększ Pomniejsz

suma2.py

```
liczba = 483284
suma = 0
while liczba > 0:
    # ostatnia cyfra
    suma = suma + liczba % 10
    # liczba bez ostatniej cyfry
    liczba = liczba // 10
print(suma)
```

Wykonywanie: suma2.py

```
29
>>>
```

Rys. 5. Wykorzystanie pętli **`while`** do liczenia sumy cyfr

DZIAŁANIA WARUNKOWE

Aby w programie pisanym w Pythonie wybrać określoną akcję w zależności od wartości zmiennej lub wyrażenia, należy zastosować instrukcję warunkową. Instrukcja ta składa się z trzech części: **`if`** (jeżeli), **`elif`** (jeżeli natomiast) oraz **`else`** (w przeciwnym przypadku). Tylko pierwsza z nich jest obowiązkowa.

```
if <warunek>:
    <instrukcja>
elif <warunek>:
    <instrukcja>
else:
    <instrukcja>
```

Rys. 6. Składnia instrukcji warunkowej

- Przeanalizuj poniższy program określający, czy dana liczba jest dodatnia, ujemna czy też równa 0 – zwróć uwagę na dwukropek kończący warunki **`if`**, **`elif`** i **`else`** oraz wcięcia przed instrukcjami dla danego warunku. Sprawdź poprawność komunikatów dla innych danych.

Tryb Nowy Otwórz Zapisz Zatrzymaj Debuguj REPL Plotter Powiększ Pomniejsz Motyw

dodatnia.py

```
liczba = 14
if liczba > 0:
    print("liczba dodatnia")
elif liczba == 0:
    print("zero")
else:
    print("liczba ujemna")
```

Wykonywanie: dodatnia.py

```
liczba dodatnia
>>>
```

Rys. 7. Program określający, czy dana liczba jest dodatnia, ujemna czy też równa 0

Różne oblicza znaku równości

Znak równości **=** oznacza przypisanie (nadanie wartości), podwójny znak równości **==** oznacza porównanie (sprawdzenie, czy wartości są sobie równe), a znak równości poprzedzony wykrzyknikiem **!=** służy do sprawdzania, czy wartości nie są sobie równe.

A teraz napisz program, który dla podanej liczby wyświetli napis **`parzysta`**, jeśli liczba jest parzysta, a w przeciwnym przypadku – napis **`nieparzysta`**.

- Przeanalizuj problem: parzystość liczby bada się przez sprawdzenie, czy liczba jest podzielna przez 2.
- Zadeklaruj zmienną przechowującą przetwarzaną liczbę.
- Zapisz instrukcję warunkową – jeżeli liczba dzieli się przez 2 bez reszty, wypisz na ekranie „parzysta".
- W przeciwnym wypadku wypisz na ekranie „nieparzysta".



Rys. 8. Program sprawdzający parzystość liczby

ZADANIA

1. Przeanalizuj poniższy skrypt i zastanów się, co będzie wypisane na ekranie po jego uruchomieniu.

```
ile_lat = 15
if ile_lat > 18:
    print("jesteś osobą pełnoletnią")
elif ile_lat == 18:
    print("od tego roku jesteś osobą pełnoletnią")
else:
     print("jesteś osobą niepełnoletnią")
```

2. Napisz program, który dla podanej liczby całkowitej wypisze komunikat, czy podana liczba jest podzielna przez 3.

3. Napisz program, który dla podanej liczby całkowitej wypisze iloczyn tej liczby przez 2 w przypadku, gdy podana liczba jest nieparzysta, lub jej kwadrat, gdy jest parzysta. Gdy liczba jest równa 0, powinien pojawiać się napis „zero”.

2.6 Odgadniesz liczbę?

DOWIESZ SIĘ, JAK

- wyszukiwać element w zbiorze uporządkowanym,
- stosować wyszukiwanie binarne według metody „dziel i zwyciężaj”,
- losować liczby całkowite.

Ostatnia lekcja ma w dużej mierze charakter powtórzeniowy. Tym razem stworzysz w Pythonie grę *Odgadnij liczbę*, w której użytkownik będzie odgadywał liczbę wylosowaną przez komputer. Do dzieła!

WYSZUKIWANIE BINARNE

Wyobraź sobie, że komputer wylosował liczbę z zakresu od 1 do 100, a ty masz odgadnąć, jaka to liczba. Podajesz swoją propozycję, a komputer ci odpowiada:

- trafiona (jeśli odgadniesz wylosowaną przez niego liczbę),
- za mała (jeśli podana przez ciebie liczba jest mniejsza niż wylosowana),
- za duża (jeśli podana przez ciebie liczba jest większa niż wylosowana).

Gdy odgadujesz liczbę z zakresu od 1 do 100, możesz trafić za pierwszym razem albo dopiero za setnym. Metoda przypadkowego wyszukiwania liczb nie jest ani dobra, ani szybka. Co możesz zrobić, by odgadnąć liczbę w najmniejszej liczbie prób? Zastosuj wyszukiwanie binarne według metody „dziel i zwyciężaj”.

Rozpocznij od elementu środkowego badanego zbioru. W przypadku liczb od 1 do 100 będzie to liczba 50 lub 51. Sprawdź, czy jest to liczba wylosowana przez komputer. Jeśli tak, to problem został rozwiązany. Jeśli nie, to poszukiwany element jest albo mniejszy od elementu środkowego, albo większy. Ponieważ zbiór liczb jest uporządkowany, elementy mniejsze od środkowego będą leżały w pierwszej połówce zbioru, a elementy większe – w drugiej. Kontynuuj poszukiwania w coraz mniejszych zbiorach, aż znajdziesz wylosowaną liczbę.

W zbiorze stuelementowym trzeba wykonać maksymalnie siedem kroków:

1. Trafiasz w połowę zakresu, zmniejszasz zbiór do 50 elementów.
2. Trafiasz w połowę zakresu, zmniejszasz zbiór do 25 elementów.
3. Trafiasz w połowę zakresu, zmniejszasz zbiór do 13 elementów.
4. Trafiasz w połowę zakresu, zmniejszasz zbiór do 7 elementów.
5. Trafiasz w połowę zakresu, zmniejszasz zbiór do 4 elementów.
6. Trafiasz w połowę zakresu, zmniejszasz zbiór do 2 elementów.
7. Pozostał ostatni element.

LOSOWANIE LICZB

A teraz sprawdź, jak działa losowanie liczb w Pythonie.

- Zaimportuj funkcję **randint** z modułu **random** odpowiadającego za losowanie. Funkcja ma dwa parametry, odpowiadające za zakres losowanych liczb.
- Uruchom kilka razy poniższy program losowania liczby z zakresu od 1 do 100. Zwróć uwagę na to, jak zmieniają się wylosowane liczby.

Tryb Nowy Otwórz Zapisz Zatrzymaj Debuguj REPL Plotter

losowanie.py

```
from random import randint
los = randint(1,100)
print(los)
```

Wykonywanie: losowanie.py

```
54
>>>
```

Rys. 1. Losowanie liczby z zakresu od 1 do 100 i przykładowy wynik

GRA W PYTHONIE

To wprowadzenie i umiejętności zdobyte podczas poprzednich lekcji powinny ci wystarczyć, by stworzyć grę, w której użytkownik będzie odgadywał liczbę z zakresu od 1 do 100 wylosowaną przez komputer.

- Zaimportuj funkcję **randint** z modułu **random**, aby przeprowadzić losowanie. Funkcja **randint(a,b)** daje w wyniku losową liczbą całkowitą z zakresu od **a** do **b**.
- Zainicjuj losowanie liczby przez komputer – zadeklaruj zmienną przechowującą liczbę z zakresu od 1 do 100 wylosowaną przez komputer, np. **los**.

```
from random import randint
los = randint(1,100)
```

Rys. 2. Import funkcji **randint** oraz inicjowanie losowania liczby przez komputer

- Zainicjuj wczytywanie liczby podanej przez gracza – niech zostanie zapamiętana np. jako **odp**. Zastosuj funkcję **int()** podczas wczytywania odpowiedzi, aby podana wartość była liczbą całkowitą.

```
print("Odgadnij liczbę od 1 do 100. ")
odp = int(input("Podaj liczbę!"))
```

Rys. 3. Inicjowanie wczytywania liczby podanej przez gracza

- Sprawdź, czy liczby **los** i **odp** są sobie równe – zastosuj pętlę **while**.

- Jeśli liczby są różne, to wykonywane są instrukcje pętli – wyświetla się komunikat **`Za duża!`** lub **`Za mała!`** i gracz zostaje poproszony o podanie następnej liczby.

```
while odp != los:
    if odp > los:
        print("Za duża!")
    else:
        print("Za mała!")
    odp = int(input("Podaj liczbę! "))
```

Rys. 4. Pętla **`while`** z instrukcją warunkową

► Jeśli liczby są sobie równe, gra się kończy – wyświetla się komunikat **`Trafiona!`**.

```
print("Trafiona!")
```

Rys. 5. Komunikat na zakończenie gry

Inne edytory

Podczas dalszej nauki programowania w języku Python możesz korzystać z edytora Mu albo z IDLE – domyślnego zintegrowanego edytora języka programowania Python (do pobrania ze strony **https://www.python.org/downloads**) lub też z edytora online Trinket.io (wymagana jest rejestracja).

ZADANIA

1. Przeanalizuj, ile razy co najmniej musisz odgadywać liczbę, gdy przedział liczb jest złożony z miliona elementów. Możesz do tego wykorzystać arkusz kalkulacyjny.

2. Zastanów się, czy taką samą strategię można zastosować w zbiorze nieuporządkowanym. Odpowiedź uzasadnij.

Jak zapisać definicje funkcji w Pythonie?

Definicję funkcji rozpoczyna słowo **`def`**, następnie jest nazwa funkcji, a za nią nawias z parametrami lub pusty. Kolejne wiersze to polecenia do wykonania. Aby wywołać funkcję, wystarczy podać jej nazwę oraz nawias z wartością parametru lub pusty.





Czym różnią się pętle while i for?

Iterację, czyli powtarzanie ciągu poleceń w pętli określoną liczbę razy lub aż do spełnienia określonego warunku, można wykonać odpowiednio za pomocą pętli **`for`** oraz **`while`**. Pętla **`for`** wykonuje ciąg poleceń tyle razy, ile wynosi parametr funkcji **`range()`**.

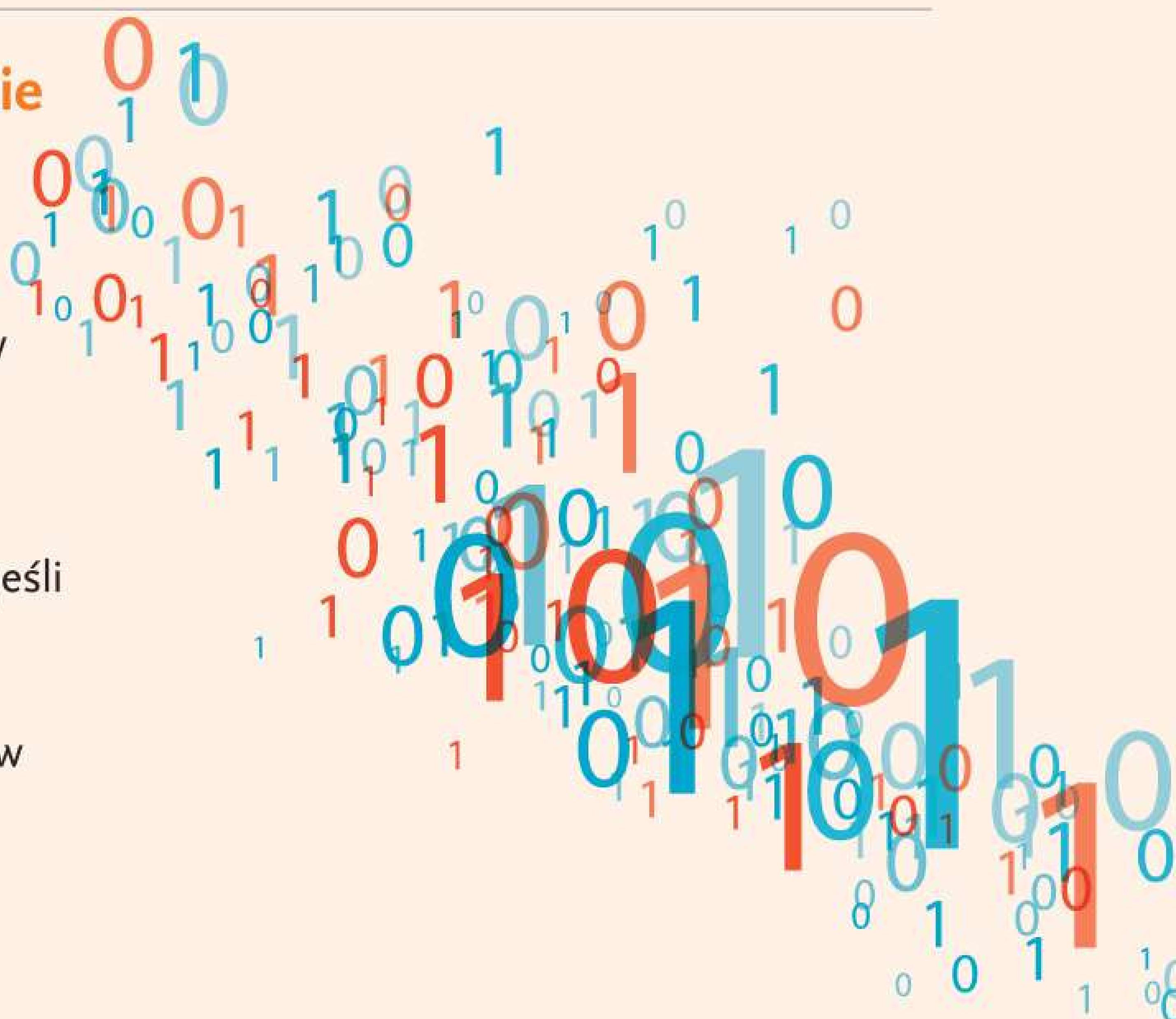
```
for i in range(3):
  fd(100); lt(120)
```

Pętla **`while`** wykonuje ciąg poleceń, dopóki podany warunek jest prawdziwy.

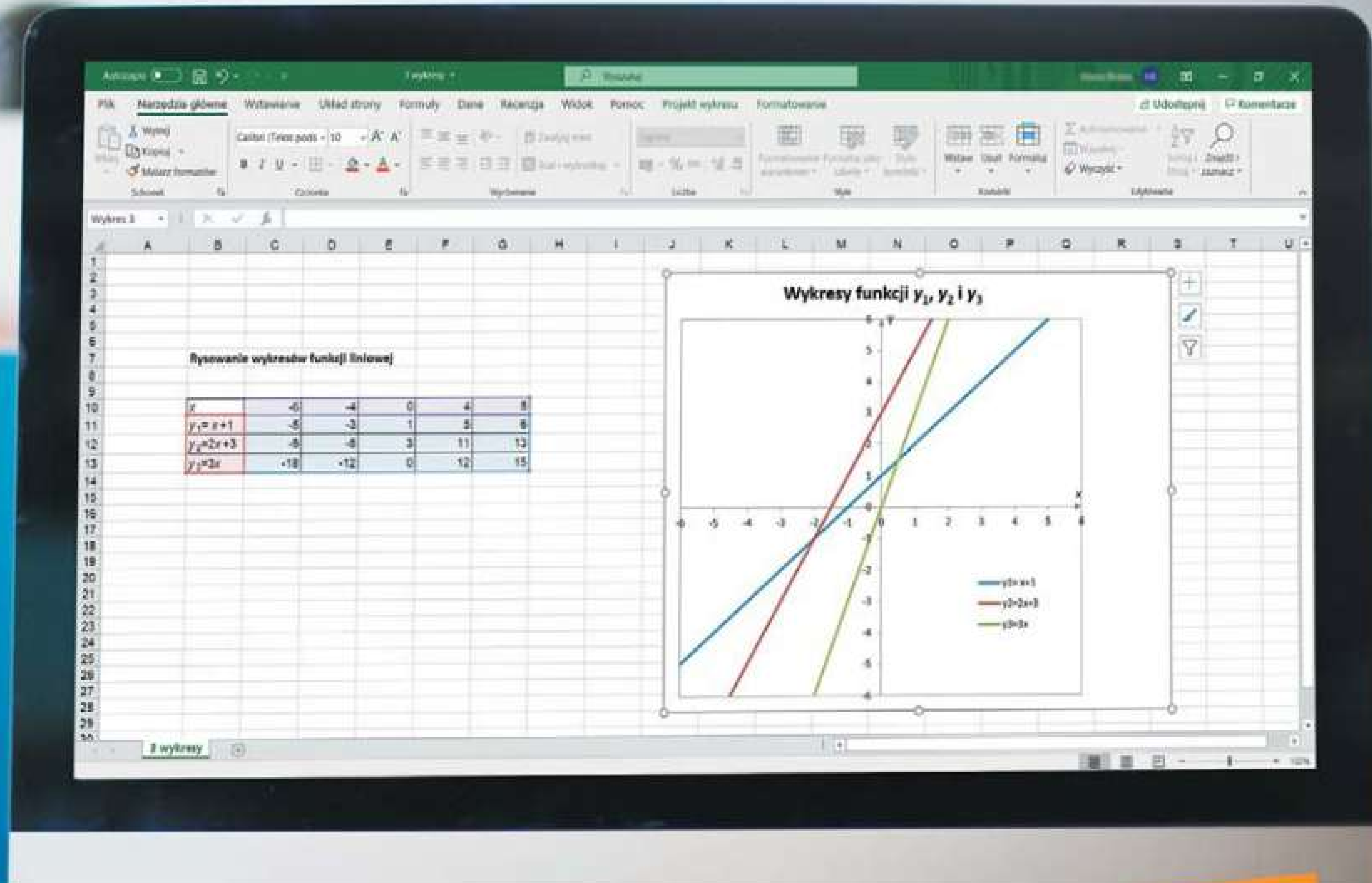
```
a = -10
while a < 0:
  a += 1
```

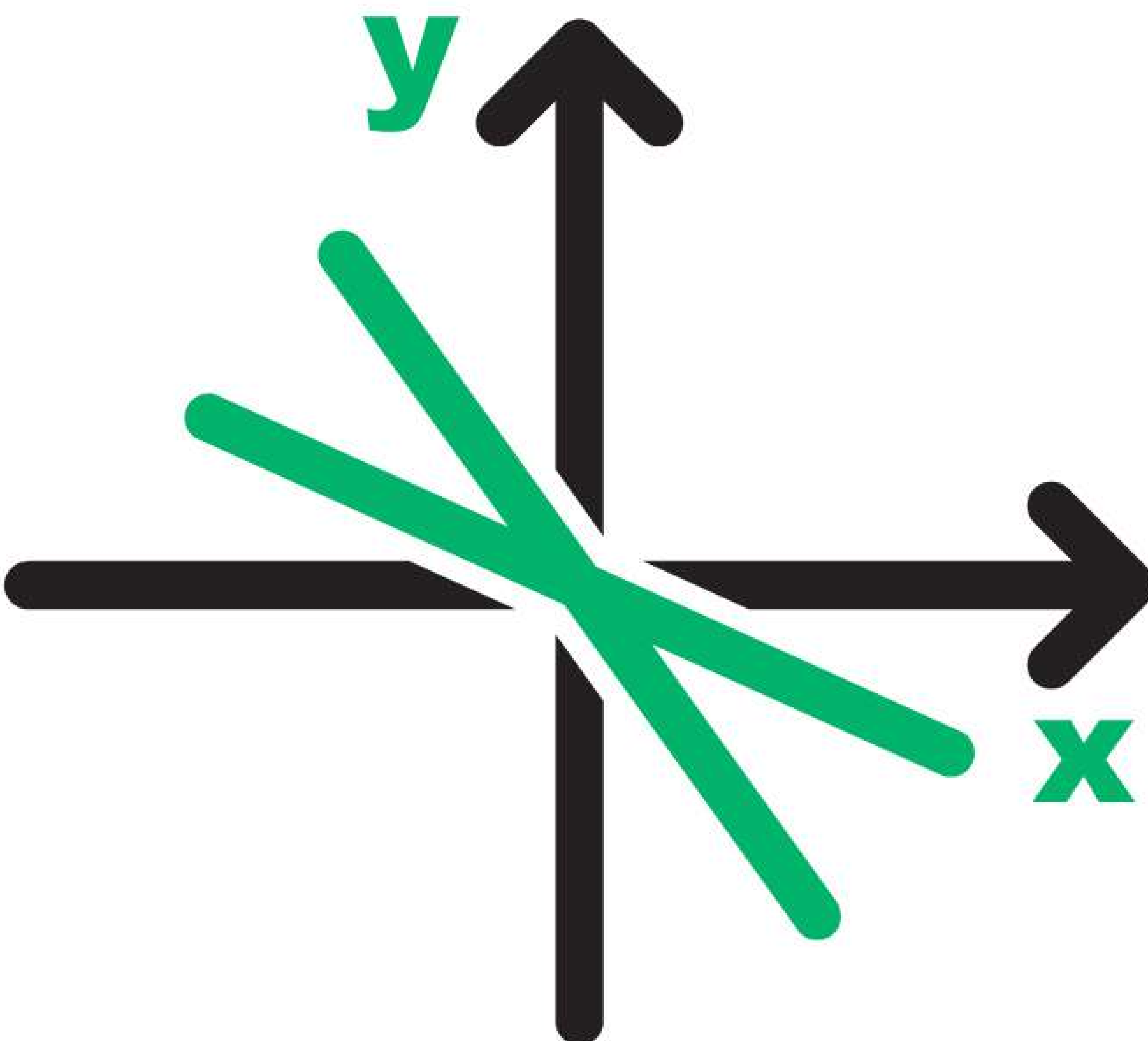
Czym różni się wyszukiwanie liniowe od wyszukiwania binarnego?

Wyszukiwanie liniowe polega na przeglądaniu kolejnych elementów i wymaga w najgorszym przypadku przejrzenia wszystkich elementów zbioru. Wyszukiwanie binarne opiera się na metodzie „dziel i zwyciężaj" – jeśli zatem uporządkowany zbiór danych ma milion elementów, to wystarczy sprawdzić maksymalnie 20 elementów w celu znalezienia żądanej wartości.



3 Lekcje z danymi





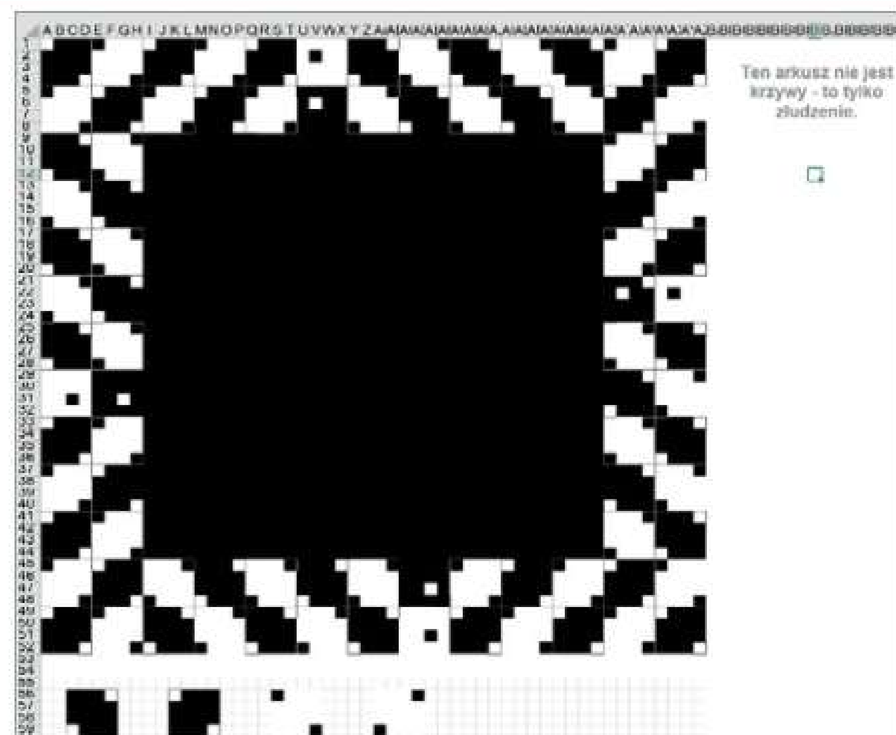
Pierwszy arkusz kalkulacyjny dla komputerów osobistych pojawił się na rynku w 1979 r. pod nazwą VisiCalc – od *visible calculator* (czytaj: wyzybyl kalkulejtor; widzialny kalkulator). Zaprojektowali go Dan Bricklin (czytaj: briklin) i Bob Frankston, absolwenci MIT, prestiżowej amerykańskiej politechniki. Arkusz liczył 254 wiersze i 63 kolumny, których nie dało się formatować. Programu nie wyposażono w funkcję rysowania wykresów ani w system pomocy.

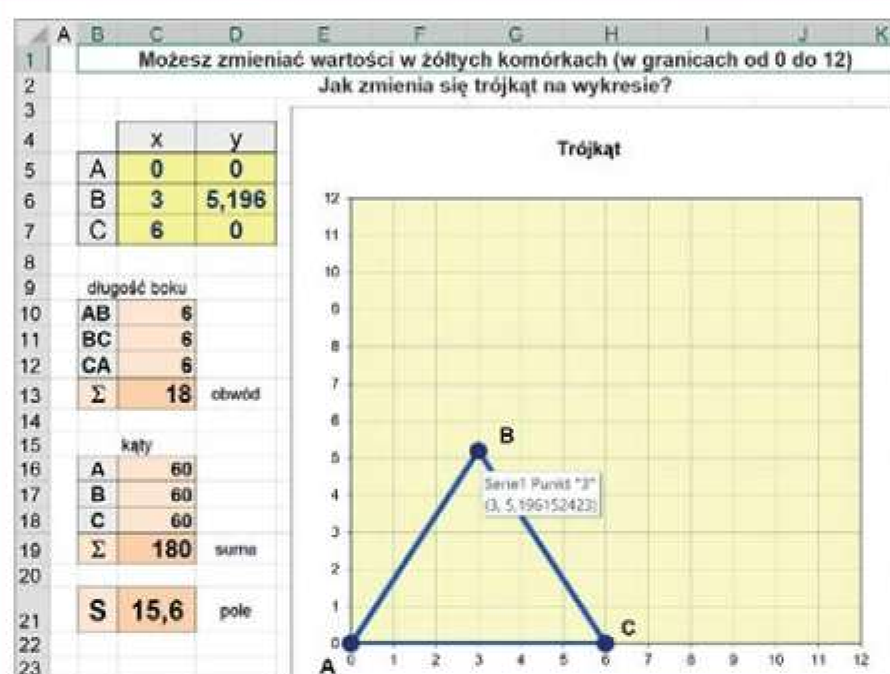
WIZUALIZACJA W ARKUSZU

Wizualizacja danych zgromadzonych w arkuszu i przedstawionych w pewnej usystematyzowanej (zazwyczaj liczbowej) postaci pozwala na łatwe i szybkie zrozumienie zależności i związków opisanych przez te dane. Do wizualizacji można wykorzystać m.in. wykresy i ilustracje, formatowanie komórek czy odpowiedni kolor czcionki. Oto kilka przykładów.

KRZYWY ARKUSZ

W tym arkuszu nie ma obliczeń, są tylko komórki z białym i czarnym tłem.



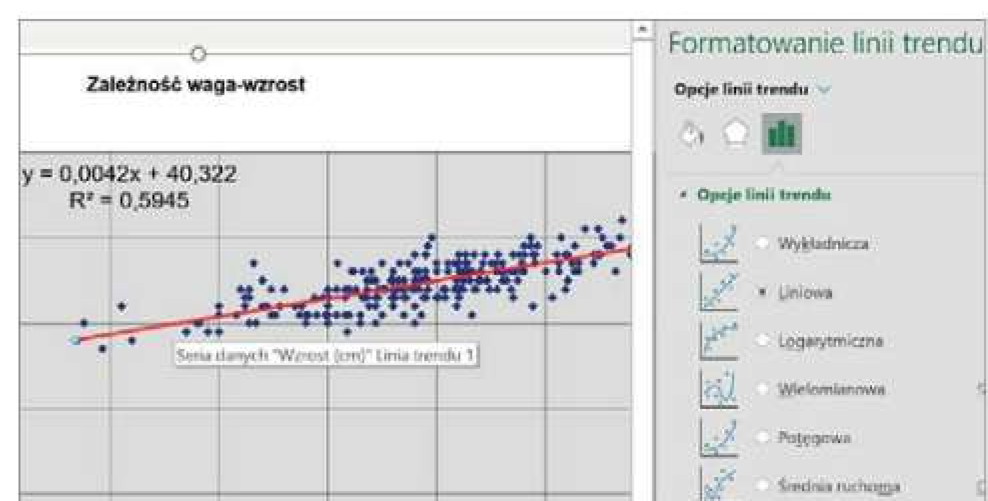


TRÓJKĄT

W żółtej tabelce można ustalać współrzędne punktów, a na wykresie obserwować, jaki trójkąt z nich powstaje.

LINIA TRENDU

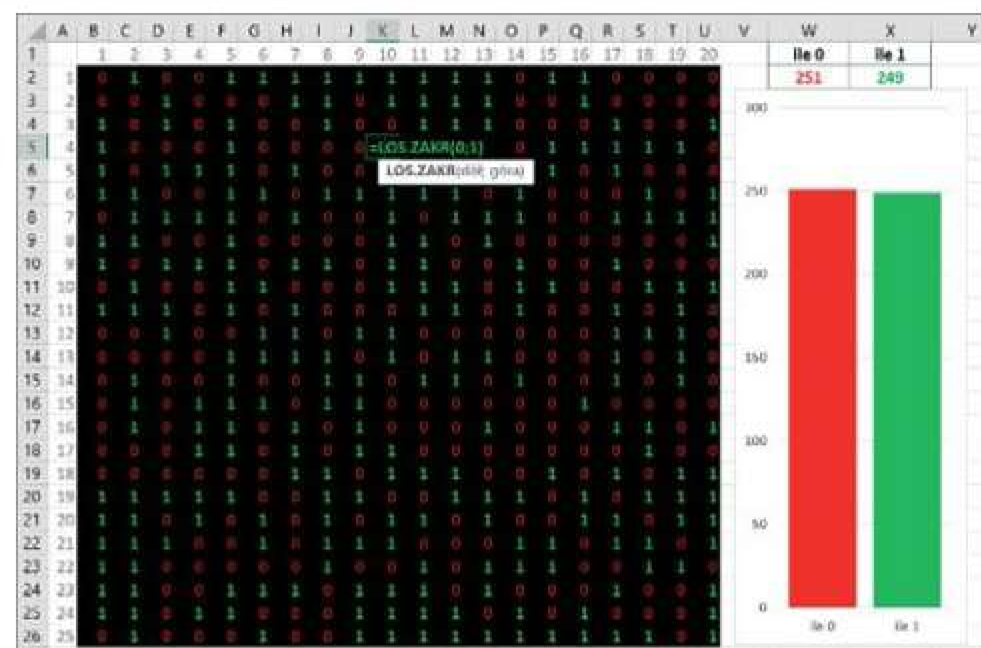
Czerwona linia na wykresie to linia trendu pokazująca zależność między dwiema mierzonymi wielkościami.

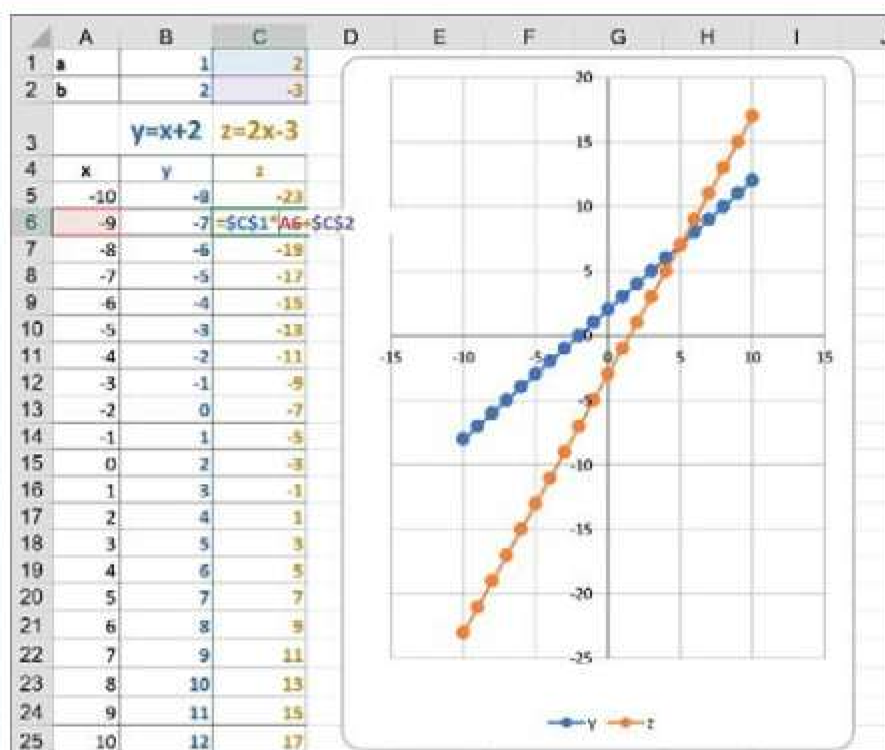


Znak drogowy, trójwymiarowy model budynku, animowana prognoza pogody, interaktywna mapa nieba, ta infografika – to wszystko przykłady wizualizacji mniej lub bardziej skomplikowanych zagadnień oraz procesów.

LOSOWANIA

Tabela zawiera pięćset losowanych przez arkusz wartości: zer i jedynek. Wykres pokazuje, ile jest jednych i drugich.



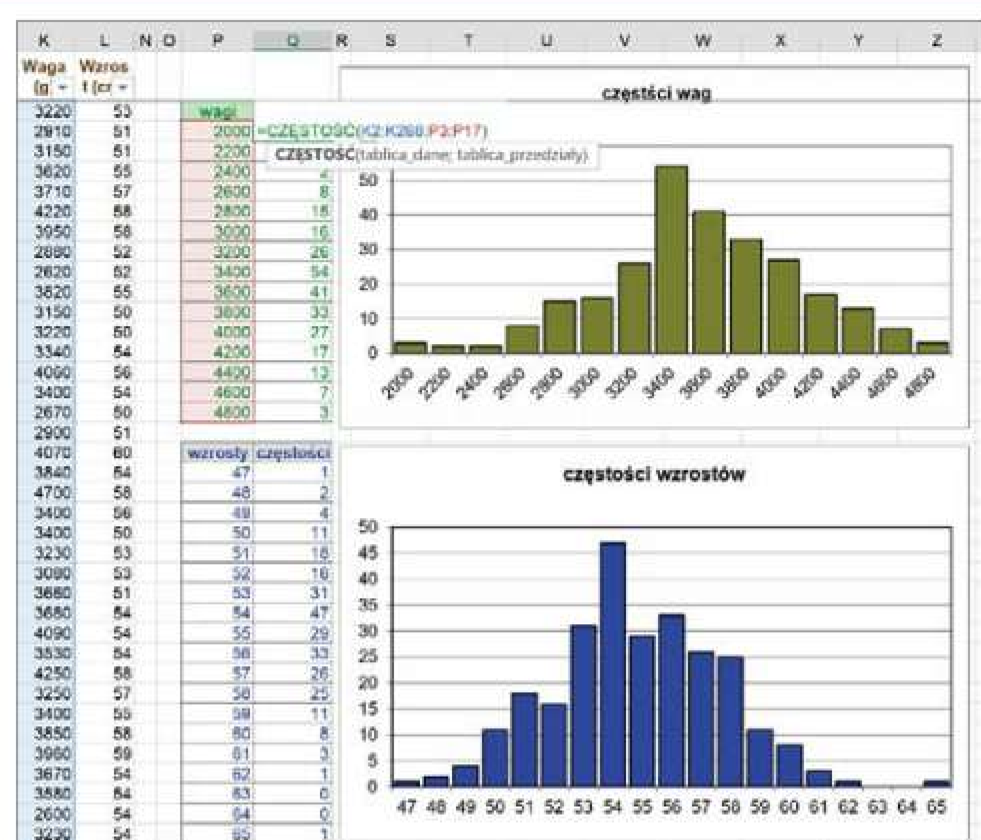


ZALEŻNOŚCI LINIOWE

Arkusz zawiera tabelę wartości dla dwóch różnych zależności liniowych i wykres punktowy XY.

HISTOGRAMY

Oba wykresy to histogramy, czyli wykresy częstości, z jaką występują poszczególne wartości odłożone na osi poziomej.



3.1 Jak to z Gaussem było

DOWIESZ SIĘ,

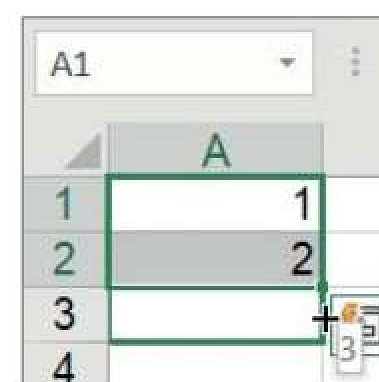
- jak sumować liczby w arkuszu kalkulacyjnym,
- jak analizować obliczenia w poszukiwaniu prawidłowości,
- czym jest ciąg arytmetyczny i jak wyznaczyć jego sumę.

Karol Fryderyk Gauss (1777–1855), niemiecki matematyk, fizyk, astronom i geodeta, już jako dziecko wykazywał zdolności do nauk ścisłych. Jak mówi anegdota, podczas jednej z lekcji matematyki był tak niesforny, że za karę dostał pracochłonne zadanie: miał obliczyć sumę 100 kolejnych liczb naturalnych, od jedynki poczynając. Karol bardzo szybko podał wynik – zauważył pewną zależność i sumę wyznaczył w pamięci.

SZYBKIE SUMOWANIE

Jak dziś szybko obliczyć sumę kolejnych liczb naturalnych od 1 do 100? Arkusz kalkulacyjny zrobi to bardzo sprawnie.

W kolumnie **A** arkusza umieść liczby od 1 do 100. Aby ułatwić sobie pracę, wpisz dwie pierwsze liczby tworzące serię, a do wprowadzenia następnych zastosuj metodę serii danych – zaznacz obie wypełnione komórki, chwyć za uchwyt wypełniania i wciąż wciskając lewy przycisk myszy, wolno przeciągnij ten uchwyt w dół, aż do setnego wiersza, tak by kolumna wypełniła się liczbami.



Rys. 1. Zaznaczenie i uchwyt wypełniania

Matematyk powiedziałby, że w arkuszu powstał w ten sposób ciąg arytmetyczny – uporządkowany zbiór liczb, w którym każdy następny wyraz jest sumą wyrazu poprzedniego oraz ustalonej liczby zwanej różnicą ciągu. W tym ciągu pierwszy wyraz i różnica między kolejnymi wyrazami wynoszą 1.

Co to jest formuła?

Formuła to wzór umieszczony w komórce arkusza. Zaczyna się od znaku równości, dalej znajduje się zapis obliczeń do wykonania lub nazwa funkcji (np. **SUMA**) oraz nawias z argumentami funkcji (liczbami lub zakresem komórek, np. **B2:B30**). Jeśli klikniesz komórkę z wpisaną formułą, formuła wyświetli się w polu edycji.

▶ Ustaw kursor w komórce **A101** i na pasku narzędzi wybierz przycisk ∑ **Autosumowanie**. Zaznaczona zostanie cała kolumna wypełniona liczbami. Formułę, **=SUMA(A1:A100)**, która pojawi się w komórce, zatwierdź klawiszem **Enter**.

SUMA | =SUMA(A1:A100)

	A	B	C	D
96	96			
97	97			
98	98			
99	99			
100	100			
101	=SUMA(A1:A100)			
102	SUMA(**liczba1**; [liczba2]; ...)			
103				

Rys. 2. Formuła sumowania

Gotowe – w komórce **A101** znajduje się suma stu kolejnych liczb.

ANALIZOWANIE OBLICZEŃ I PRAWIDŁOWOŚCI

A teraz zbuduj arkusz do sumowania liczb i na jego podstawie wyprowadź wzór wymyślony przez Gaussa.

Metoda 1

▶ Przejdź do drugiego arkusza i utwórz tabelę złożoną z trzech kolumn.

- Kolumnę **A** nazwij ***n* (rosnąco)** i wpisz w nią kolejne liczby naturalne od 1 do 100.
- Kolumnę **B** nazwij ***n* (malejąco)** i uzupełnij ją w odwrotnej kolejności.
- Kolumnę **C** nazwij **suma**.

	A	B	C
1	***n* (rosnąco)**	***n* (malejąco)**	**suma**
2	1	100	
3	2	99	
4	3	98	
5	4	97	
6	5	96	
7	6	95	
8	7	94	
9	8	93	

Rys. 3. Częściowo uzupełniony arkusz z nagłówkami – metoda 1

▶ Wypełnij kolumnę **C** kolejnymi sumami par liczb w wierszach – ustaw kursor w komórce **C2**, kliknij ∑ **Autosumowanie**, zaakceptuj wynik **=SUMA(A2:B2)** klawiszem **Enter**, a następnie skopiuj ten wzór do kolejnych komórek i sprawdź wyniki.

- Przeanalizuj tabelę – sprawdź, ile wierszy zostało zsumowanych i jakie są sumy we wszystkich wierszach.

- Oblicz sumę wszystkich sum w komórce **C102** i porównaj ją z sumą liczb od 1 do 100 – w tabeli zsumowano *n* wierszy, suma w każdym wierszu ma wartość $n + 1$, a suma wszystkich sum jest dwukrotnie większa od sumy, jaką należało obliczyć.

Czy teraz potrafisz już podać wzór Gaussa na obliczenie sumy kolejnych *n* liczb?

Sposoby na sumowanie

Formułę **=SUMA()** można wpisać z klawiatury albo wybrać na karcie **Narzędzia główne** w grupie **Edytowanie** za pomocą przycisku Σ **Autosumowanie** lub ikony f_x na pasku formuły. W przypadku sumowania wartości z danego zakresu oraz wartości z pojedynczych komórek należy kliknąć przycisk Σ **Autosumowanie** pod zakresem komórek i z wciśniętym klawiszem **Ctrl** kliknąć wybrane pojedyncze komórki, a potem zatwierdzić klawiszem **Enter**.

Metoda 2

Do wzoru Gaussa można dojść także inną metodą. Przekonasz się o tym, gdy przeanalizujesz liczby naturalne z zakresu 1–20.

- Przejdź do kolejnego arkusza w swoim skoroszycie i utwórz tabelę złożoną z sześciu kolumn.
  - Kolumnę **A** nazwij ***n*** i uzupełnij ją kolejnymi liczbami naturalnymi od 1 do 20.
  - Kolumnę **B** nazwij **suma *n*** (uzupełnisz ją sumami kolejnych liczb – od 1 do liczby w bieżącym wierszu).
  - Kolumnę **C** nazwij **2*n*** i uzupełnij ją kolejnymi liczbami parzystymi.
  - Kolumnę **D** nazwij **suma 2*n*** (uzupełnisz ją sumami kolejnych liczb – od 2 do liczby w bieżącym wierszu).
  - Kolumnę **E** nazwij **2*n* – 1** i uzupełnij ją kolejnymi liczbami nieparzystymi.
  - Kolumnę **F** nazwij **suma (2*n* – 1)** (uzupełnisz ją sumami kolejnych liczb – od 1 do liczby w bieżącym wierszu).

	A	B	C	D	E	F
1	*n*	suma *n*	2*n*	suma 2*n*	2*n* - 1	suma (2*n* - 1)
2	1		2		1	
3	2		4		3	
4	3		6		5	
5	4		8		7	
6	5		10		9	
7	6		12		11	
8	7		14		13	

Rys. 4. Częściowo uzupełniony arkusz z nagłówkami – metoda 2

- Wypełnij kolumnę **B** kolejnymi sumami liczb naturalnych.
  - Ustaw kursor w komórce **B2**, wybierz ∑ **Autosumowanie** i zaakceptuj wynik **=SUMA(A2)**.
  - Zrób to samo w komórce **B3**. Tym razem trzeba „poprawić" wynik podany przez arkusz, zaznaczając myszą zakres sumowania od **A2** do **A3**.
  - Skopiuj wzór do następnej komórki (zaznacz **B3** i za pomocą uchwytu wypełniania przeciągnij na **B4**) i sprawdź wynik. Niestety, nie jest on poprawny – zostały zsumowane dwie liczby z komórek **A3** i **A4**. Arkusz „nie domyślił się", że sumowanie chcesz zaczynać zawsze od komórki **A2**. Dzieje się tak, ponieważ podczas kopiowania arkusz traktuje adresy komórek w sposób względny: jeśli w komórce **B3** była **SUMA(A2:A3)**, to w komórce **B4** będzie **SUMA(A3:A4).** Aby zadeklarować, że adres **A2** ma być traktowany bezwzględnie jednakowo (nie podlegać zmianie), trzeba wstawić przed numerami kolumny i wiersza znak **$**. Adres bezwzględny należy więc zapisać jako **A2** (choć w tym wypadku wystarczy **A$2** – czy wiesz dlaczego?).
  - Popraw wzór w komórce **B3** na **=SUMA(A$2:A3)** i skopiuj go do pozostałych komórek w kolumnie **B**.
- Kolumny **D** i **F** uzupełnij w podobny sposób. Pamiętaj, żeby sprawdzać uzyskane wyniki – podczas wpisywania wartości czy formuł łatwo popełnić błąd.

Twoja tabela powinna być podobna do przedstawionej poniżej.

	A	B	C	D	E	F
1	*n*	**suma *n***	**2*n***	**suma 2*n***	**2*n* - 1**	**suma (2*n* - 1)**
2	1	**1**	2	**2**	1	**1**
3	2	**3**	4	**6**	3	**4**
4	3	**6**	6	**12**	5	**9**
5	4	**10**	8	**20**	7	**16**
6	5	**15**	10	**30**	9	**25**
7	6	**21**	12	**42**	11	**36**
8	7	**28**	14	**56**	13	**49**
9	8	**36**	16	**72**	15	**64**
10	9	**45**	18	**90**	17	**81**
11	10	**55**	20	**110**	19	**100**
12	11	**66**	22	**132**	21	**121**
13	12	**78**	24	**156**	23	**144**
14	13	**91**	26	**182**	25	**169**
15	14	**105**	28	**210**	27	**196**
16	15	**120**	30	**240**	29	**225**
17	16	**136**	32	**272**	31	**256**
18	17	**153**	34	**306**	33	**289**
19	18	**171**	36	**342**	35	**324**
20	19	**190**	38	**380**	37	**361**
21	20	**210**	40	**420**	39	**400**
22	21	**231**	42	**462**	41	**441**
23	22	**253**	44	**506**	43	**484**

Rys. 5. Uzupełniona tabela – metoda 2

Na koniec rozbuduj tabelę tak, aby obejmowała wszystkie liczby parzyste i nieparzyste od 1 do 100.

Zauważ, że w kolumnach **A**, **C** i **E** znajdują się ciągi o takiej własności, że różnica między kolejnymi wyrazami jest jednakowa; w przypadku liczb naturalnych wynosi 1, a w przypadku liczb parzystych i nieparzystych – 2.

Jak obliczyć sumę n kolejnych wyrazów takiego ciągu? Dodaj pierwszą liczbę parzystą i ostatnią liczbę nieparzystą, drugą parzystą i przedostatnią nieparzystą itd. Jaki jest wynik każdego dodawania? Ile działań musisz wykonać? Czy już wiesz, jaki to wzór? Czy jest taki sam jak ten wyprowadzony na podstawie poprzedniej tabeli?

ZADANIA

1. Oblicz w arkuszu kolejne sumy pierwszych stu liczb parzystych. Pierwszą sumą będzie 2, drugą 6. A setną?

2. Oblicz w arkuszu kolejne sumy pierwszych stu liczb nieparzystych. Pierwszą sumą będzie 1, drugą 4. A setną?

3.2 Liczby, potęgi, ciągi

DOWIESZ SIĘ, JAK
- utworzyć kalkulator potęg i ciągów,
- chronić arkusz przed zmianą danych,
- porównywać ciągi liczbowe.

Jak już wiesz, arkusz kalkulacyjny to świetne narzędzie do analizowania obliczeń i szukania prawidłowości. Bardzo przydatny jest też do tworzenia kalkulatorów ułatwiających obliczenia.

KALKULATOR POTĘG I CIĄGÓW

W arkuszu kalkulacyjnym można wprowadzać mniej i bardziej skomplikowane formuły i kopiować je do sąsiadujących komórek. To sprawia, że może on z powodzeniem zastąpić kalkulator.

Stwórz swój arkuszowy kalkulator. Zacznij od utworzenia ciągu potęg liczby 2, tj. zbioru kolejnych liczb: 2^0, 2^1, 2^2, 2^3, 2^4, 2^5...

- Utwórz nagłówki tabeli: pierwszą kolumnę nazwij ***n***, a drugą – **potęgi**.
- Uzupełnij kolumnę ***n*** – zastosuj metodę serii danych, aby wstawić liczby od 1 do 50.
- Uzupełnij kolumnę **potęgi** tak, aby znalazły się w niej kolejne potęgi dwójki uzyskane według podanej formuły.
  - Do komórki **B2** wpisz wartość 2, a do komórki **B3** – formułę obliczania kolejnych potęg. Formułę zacznij od znaku równości, następnie wpisz podstawę potęgi (**B$2**), symbol oznaczający podnoszenie do potęgi (^) oraz wykładnik potęgi (**A3**).
  - Pamiętaj o wstawieniu przed dwójką w adresie komórki **B2** symbolu **$** – w ten sposób podczas kopiowania formuły numer wiersza pozostanie niezmieniony (podstawą potęgi będzie liczba wpisana w komórce **B2**).
  - Skopiuj formułę do pozostałych komórek w kolumnie **B**.

	A	B
1	***n***	**potęgi**
2	1	2
3	2	=B$2^A3

Rys. 1. Obliczanie kolejnych potęg liczby 2

► Przyjrzyj się wynikom obliczeń. Czy ciąg jest rosnący? Z jakich składa się liczb?

	A	B
23	22	4194304
24	23	8388608
25	24	16777216
26	25	33554432
27	26	67108864
28	27	1,34E+08
29	28	2,68E+08
30	29	5,37E+08
31	30	1,07E+09
32	31	2,15E+09
33	32	4,29E+09

Rys. 2. Notacja naukowa

Ciąg składa się z coraz większych liczb:

- $2^1 = 2$;
- $2^{10} = 1024$;
- $2^{20} = 1\,048\,576$;
- $2^{27} = 1{,}34E{+}08$.

1,34E+08? Co to za liczba?

Od pewnego momentu Excel wyświetla bardzo duże (lub bardzo małe) liczby w notacji naukowej – w celu skrócenia zapisu takich liczb – a więc np. $134\,000\,000 = 1{,}34 \cdot 100\,000\,000 = 1{,}34 \cdot 10^8 = 1{,}34E{+}08$. Ale czy na pewno jest to wartość 2^{27}? Jeśli poszerzysz kolumnę **B** i sformatujesz zawartość komórki (wybierzesz z grupy **Liczba** format **Liczbowe**), zobaczysz dokładną wartość: 134 217 728. Excel liczy poprawnie, ale nie wyświetla dokładnego wyniku, gdy ma za mało miejsca w komórce. W pamięci przechowuje 15 cyfr znaczących, a zatem dopiero dla liczb, które mają w zapisie więcej niż 15 cyfr, zaczyna być niedokładny. Wtedy zaokrągla liczby, czyli podaje ich przybliżone wartości. Możesz to zauważyć, gdy sprawdzisz 49. i 50. potęgę dwójki. Pierwsza jest jeszcze dokładna (wynik ma 15 cyfr), kolejna już nie. Skąd to wiadomo?

► Teraz przejdź do kolumny **C** – nazwij ją **odwrotności** i odpowiednio uzupełnij. Wpisz w komórkę **C2** formułę obliczającą odwrotności potęg: **=1/B2**, i skopiuj ją do następnych komórek. Tym razem liczby są coraz mniejsze, ciąg jest malejący.

► Kolumnę **D** zatytułuj **sumy odwrotności**. Następnie do komórki **D2** skopiuj wartość z komórki **C2**, w komórce **D3** wpisz formułę **=D2+C3**, czyli do poprzedniej sumy dodaj bieżącą odwrotność. Sprawdź, że to działa, i oblicz sumy odwrotności dla wszystkich 50 wierszy.

► Przeanalizuj wyniki. Zwróć uwagę, że kolejne sumy zbliżały się do liczby 1, aby wreszcie osiągnąć wartość, której arkusz nie odróżnia od jedynki. Od pewnego miejsca wynik sumowania się nie zmienia.

	A	B	C	D
28	27	1,34E+08	7,45058E-09	0,999999993
29	28	2,68E+08	3,72529E-09	0,999999996
30	29	5,37E+08	1,86265E-09	0,999999998
31	30	1,07E+09	9,31323E-10	0,999999999
32	31	2,15E+09	4,65661E-10	1
33	32	4,29E+09	2,32831E-10	1
34	33	8,59E+09	1,16415E-10	1
35	34	1,72E+10	5,82077E-11	1

Rys. 3. Wyniki sumowania odwrotności potęg liczby 2

Możesz sprawdzić, że obliczanie kolejnych wyrazów (choćby ich było 1000) nic nie zmieni – ich suma nie przekroczy 1. Suma odwrotności potęg liczby 2 jest bowiem ograniczona, a granicę stanowi liczba 1.

OCHRONA ARKUSZA

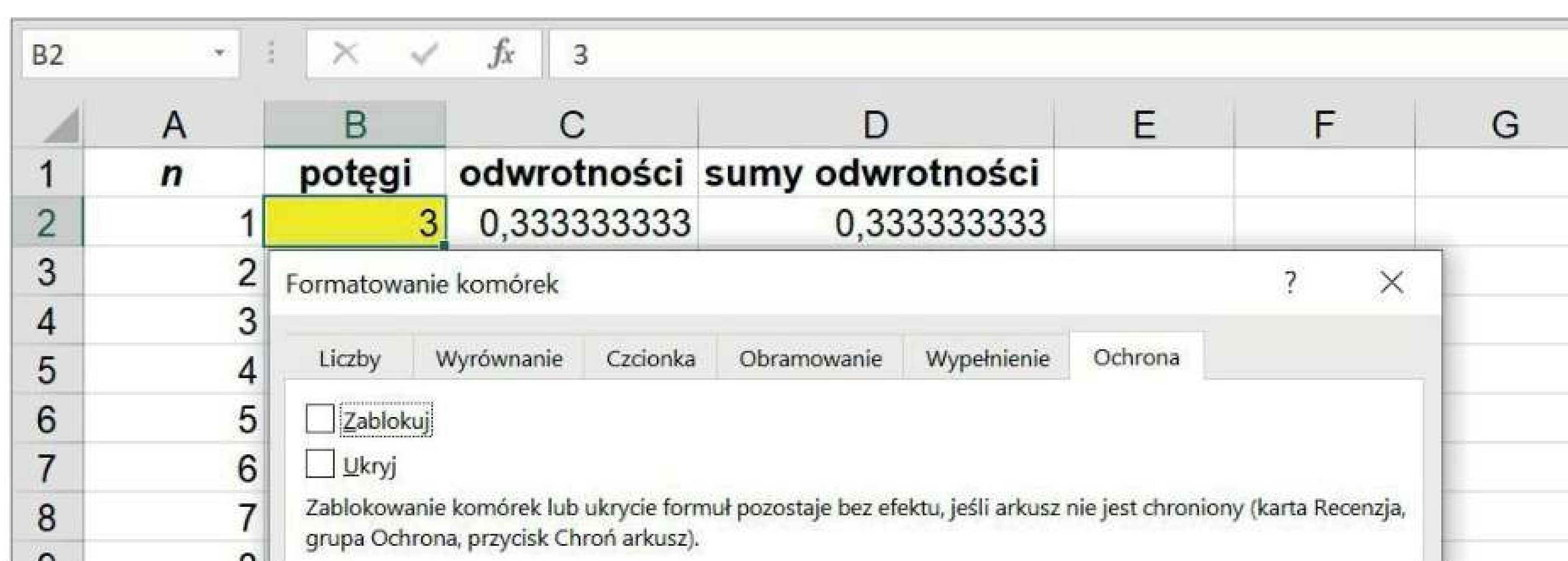
Twój kalkulator został zaplanowany tak, by wyniki obliczeń w kolumnach **B**, **C** i **D** zależały od wartości wpisanej do komórki **B2**. Oznacza to, że można w nim badać potęgi dowolnych liczb – wystarczy wpisać do komórki **B2** inną liczbę, np. 3.

	A	B	C	D
1	***n***	**potęgi**	**odwrotności**	**sumy odwrotności**
2	1	3	0,333333333	0,333333333
3	2	9	0,111111111	0,444444444

Rys. 4. Sprawdzenie działania kalkulatora

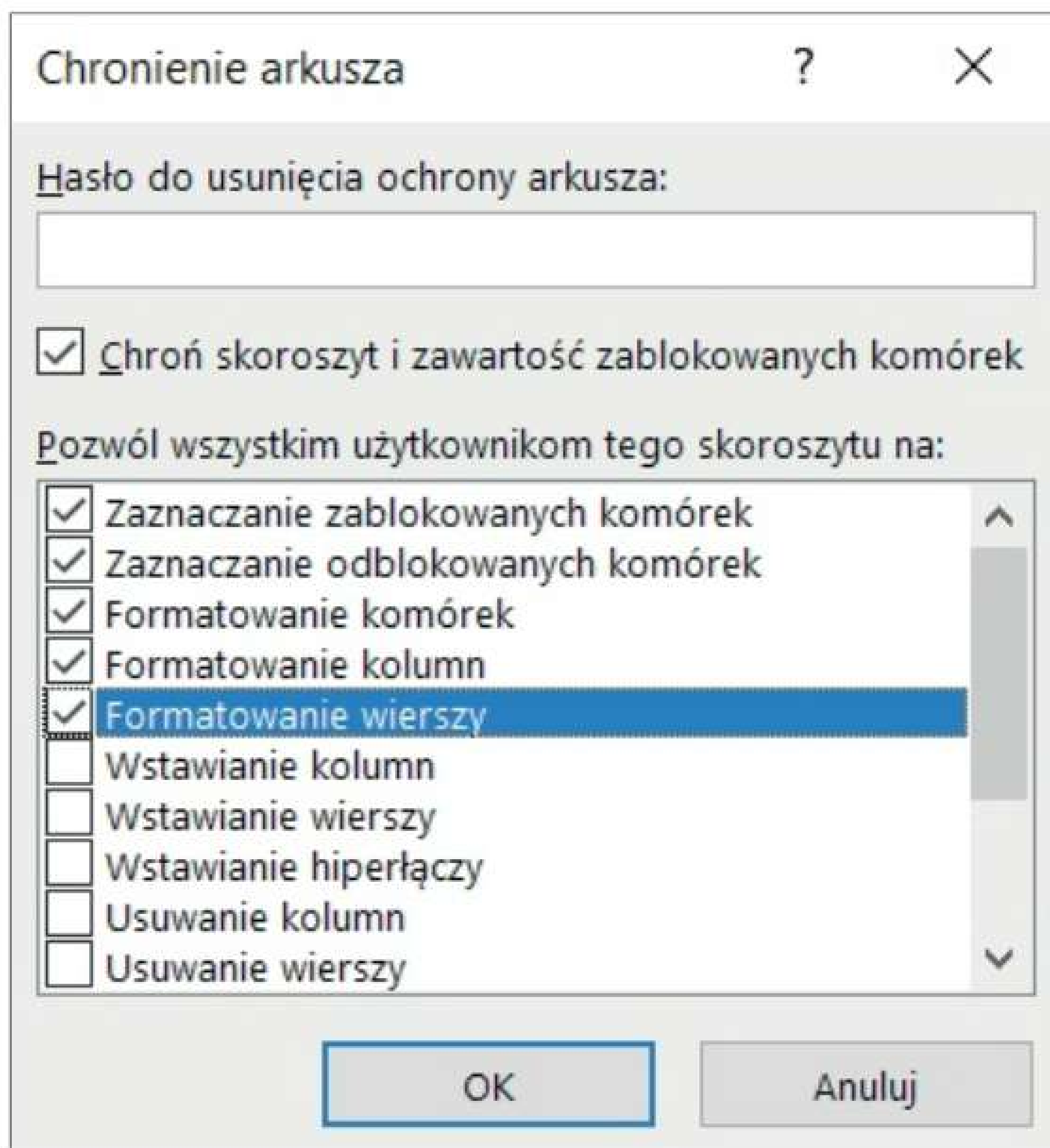
Dobrze działający kalkulator warto chronić przed przypadkowymi błędami. Aby zabezpieczyć komórki (poza komórką **B2**) przed wprowadzaniem zmian, które mogłyby zmienić działanie wprowadzonych formuł, trzeba włączyć ochronę arkusza.

- Wyróżnij komórkę **B2** – dodaj wypełnienie i krawędzie.
- Kliknij komórkę **B2** prawym przyciskiem myszy i z menu podręcznego wybierz polecenie **Formatuj komórki**.
- W oknie **Formatowanie komórek** wybierz kartę **Ochrona** i odznacz pole **Zablokuj**.



Rys. 5. Formatowanie komórek

► Włącz ochronę dla pozostałych komórek – na karcie **Recenzja** kliknij przycisk **Chroń arkusz**, a następnie w wyświetlonym oknie **Chronienie arkusza** kliknij przycisk **OK**.



Rys. 6. Włączanie ochrony arkusza

PORÓWNYWANIE CIĄGÓW

Który z ciągów rośnie najszybciej? Porównaj ciąg liczb naturalnych, ciąg powstały przez podnoszenie kolejnych liczb naturalnych do kwadratu, ciąg powstały przez podnoszenie kolejnych liczb naturalnych do trzeciej potęgi i ciąg tworzony przez obliczanie silni każdej kolejnej liczby naturalnej (silnia to iloczyn kolejnych liczb, oznaczamy ją wykrzyknikiem, np. $4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24$; aby obliczyć silnię kolejnej liczby naturalnej, można też pomnożyć przez tę liczbę silnię poprzedniej liczby, np. $5! = 4! \cdot 5$).

► Przejdź do trzeciego arkusza w swoim skoroszycie, a następnie zmień nazwę pierwszego arkusza na **potęgi**, drugiego na **co rośnie szybciej**, a trzeciego na **silnia**.

24	23	9,41E+10	1,06221E-11
25	24	2,82E+11	3,54071E-12
26	25	8,47E+11	1,18024E-12
27	26	2,54E+12	3,93412E-13
28	27	7,63E+12	1,31137E-13

potęgi | Arkusz2 | Arkusz3

Rys. 7. Nadawanie nazwy arkuszowi

► Utwórz w drugim arkuszu tabelę na wzór poniższej – uzupełnienie kolumn **A–D** nie powinno sprawić ci większego kłopotu. Kolumnę **E** pozostaw na razie pustą. Zwróć uwagę, że w tym przypadku nie trzeba podawać pierwszego wyrazu ciągu – każdy wyraz można obliczyć z podanego słownie wzoru, zaczynając od pierwszej liczby naturalnej. Wpisz liczby naturalne od 1 do 50.

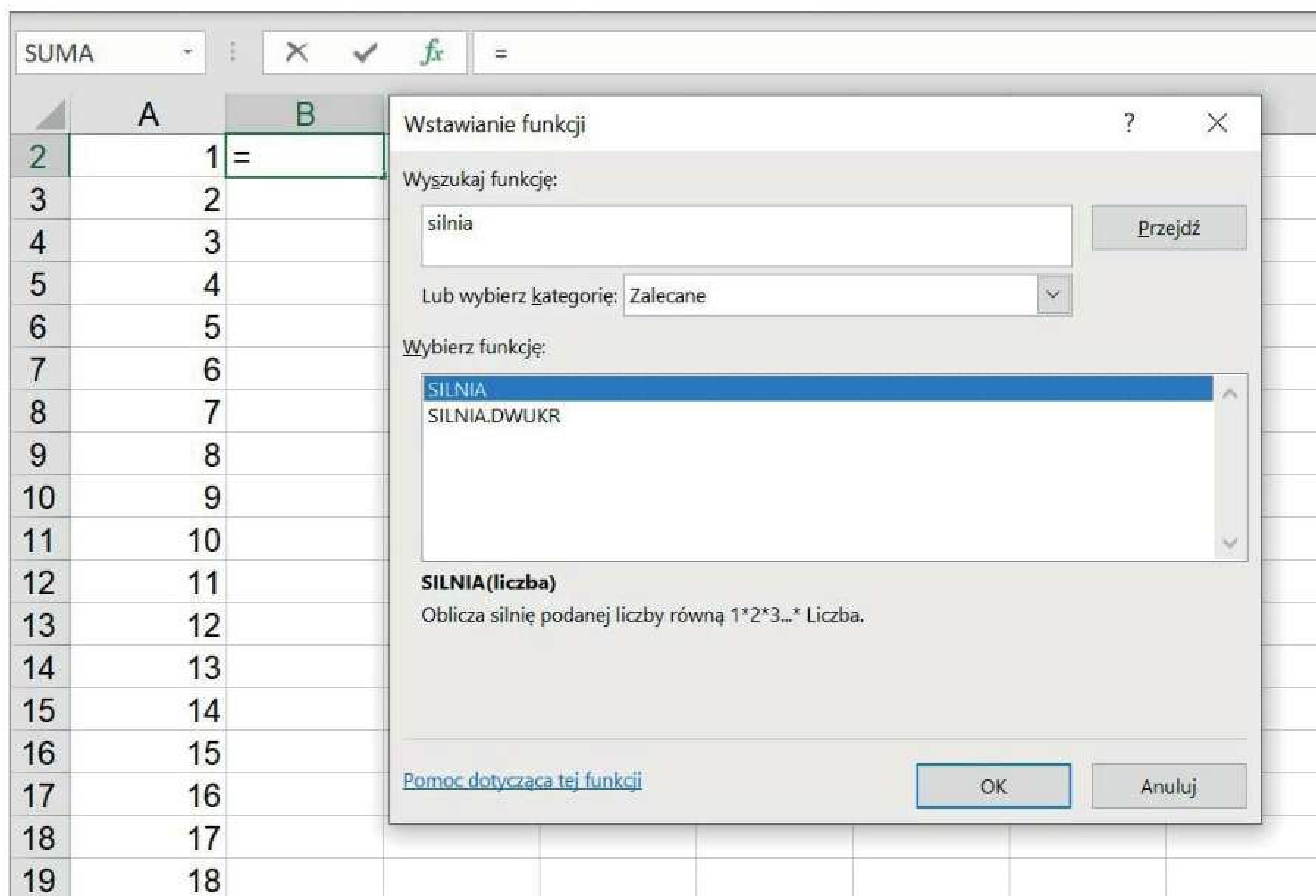
H10

	A	B	C	D	E	F
1	który wyraz ciągu	liczby naturalne	liczby naturalne do kwadratu	liczby naturalne do sześcianu	silnia liczb naturalnych	
2	1	1	1	1	1	
3	2	2	4	8		
4	3	3	9	27		
5	4	4	16	64		

Rys. 8. Tabela z kolejnymi ciągami

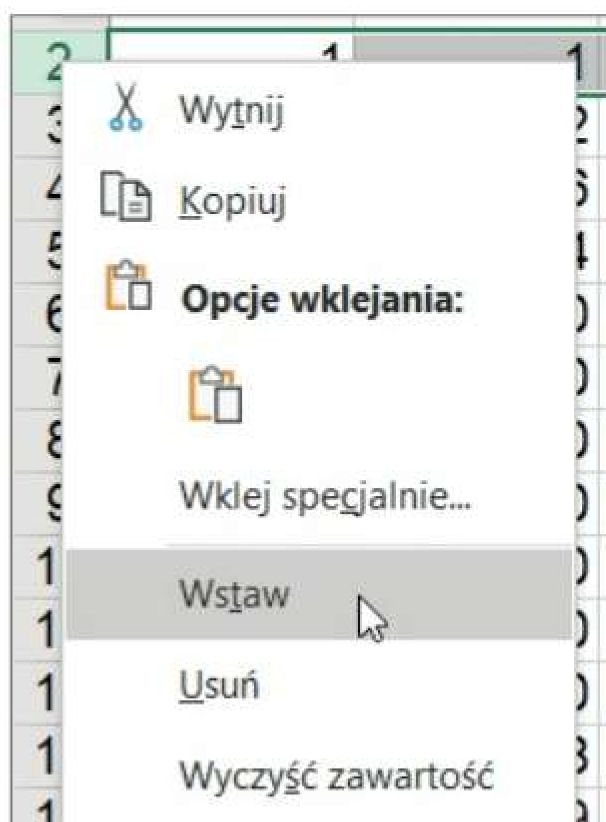
Przejdź do arkusza **silnia**.

- Kolumnę A nazwij ***n*** i uzupełnij kolejnymi liczbami naturalnymi od 1 do 50. Kolumnę **B** nazwij **silnia** – tu będziesz obliczać kolejne silnie.
- Silnia jest funkcją, więc trzeba jej poszukać wśród funkcji arkusza. Kliknij f_x na pasku formuły. W wyświetlonym oknie **Wstawianie funkcji** wpisz **silnia** i kliknij przycisk **Przejdź**.



Rys. 9. Wyszukiwanie funkcji

- W oknie **Argumenty funkcji** wpisz adres komórki **A2** i zatwierdź wstawienie funkcji przyciskiem **OK**.
- Wpisaną przez arkusz formułę **=SILNIA(A2)** skopiuj do następnych komórek.
- Sprawdź, czy można obliczyć silnię 0.
  - Wstaw dodatkowy drugi wiersz – kliknij numer wiersza prawym przyciskiem myszy i z menu podręcznego wybierz polecenie **Wstaw**.



Rys. 10. Wstawianie nowego wiersza

- Wpisz w nowym drugim wierszu w komórce **A2** wartość 0, a do komórki **B2** skopiuj formułę z komórki **B3**. Jak widzisz, 0! = 1. To kwestia umowy – tak została zdefiniowana ta funkcja.

Teraz możesz wrócić do arkusza **co rośnie szybciej** i uzupełnić kolumnę **E**. Czy już widzisz, który z badanych ciągów rośnie najszybciej?

ZADANIA

1. Dany jest ciąg o pierwszym wyrazie równym 2, zbudowany tak, że każdy kolejny wyraz powstaje przez mnożenie poprzedniego przez 2. Przygotuj arkusz, za pomocą którego obliczysz kolejne wyrazy tego ciągu. Zmodyfikuj gotowy arkusz tak, aby pierwszy wyraz ciągu można było mnożyć przez dowolną liczbę wpisaną do wybranej komórki. Wygeneruj w ten sposób 50 wyrazów ciągu.

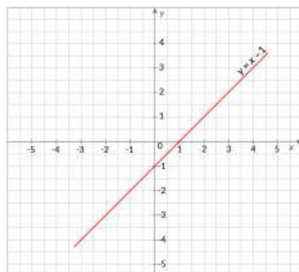
2. Dany jest ciąg, w którym kolejny wyraz otrzymuje się przez dodanie do poprzedniego wyrazu pewnej stałej liczby. Przygotuj arkusz, za pomocą którego obliczysz kolejne wyrazy ciągów o początkowych wyrazach: 2, 5, 8, 11, 14, ... oraz 120, 100, 80, 60, 40, ... Zmodyfikuj arkusz tak, aby pierwszy wyraz ciągu i dodawaną liczbę można było wpisać do wybranych komórek. Wygeneruj tak 50 wyrazów ciągu.

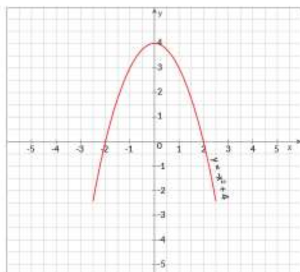
3.3 Z tabeli – wykres

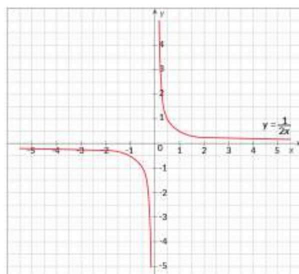
DOWIESZ SIĘ, JAK

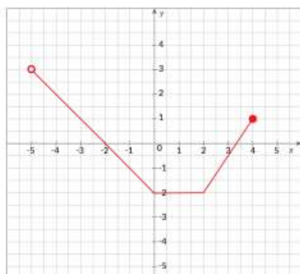
- rysować wykresy funkcji liniowych za pomocą kreatora wykresów,
- formatować wykresy funkcji,
- wprowadzać nazwy dla zakresów liczbowych w arkuszu.

Umiesz utworzyć w arkuszu kalkulacyjnym wykresy słupkowy, kolumnowy i kołowy oraz sformatować ich poszczególne elementy. W matematyce tego typu obiekty graficzne są przeważnie nazywane diagramami, a nazwa „wykres" odnosi się do wykresów funkcji, które mogą wyglądać np. tak, jak pokazano poniżej.









Rys. 1. Przykłady wykresów funkcji



WYKRESY FUNKCJI LINIOWEJ

Podczas tej lekcji zajmiesz się wykresami funkcji liniowej, którą poznasz dokładnie na matematyce dopiero w szkole ponadpodstawowej. Funkcja liniowa opisuje zależność między zmiennymi x (tzw. argumentami) i y (tzw. wartościami), np. $y = 2x$, $y = x + 5$, $y = -2x + 3$. Przykładem zależności tego typu jest zależność drogi od czasu w ruchu jednostajnym prostoliniowym. Jak się zapewne domyślasz, jej wykres ma postać linii prostej.

Na początku wykreśl prostą funkcję opisaną wzorem $y = 2x$.

- Otwórz pusty arkusz Excel i skonstruuj w nim dwuwierszową tabelę:
  - pierwszy wiersz oznacz jako ***x*** – umieść w nim argumenty (np. –2, –1, 0, 1, 2), za pomocą których na podstawie danego wzoru obliczysz wartości;
  - drugi wiersz oznacz jako ***y* = 2*x***.
- Wylicz za pomocą arkusza kolejne wartości – wprowadź nazwę dla zakresu komórek zawierających argumenty.
  - Zaznacz wszystkie komórki arkusza zawierające argumenty, kliknij pole nazwy (tuż nad kolumną **A**), wpisz w nim ***x*** i naciśnij klawisz **Enter**.

x | fx | -2

	A	B	C	D	E	F
1	***x***	-2	-1	0	1	2
2	***y=2x***					

Rys. 2. Wprowadzanie nazwy dla zakresu komórek

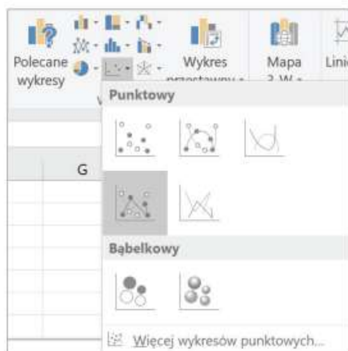
- Do komórki **B2** wstaw formułę: **=2*x** i przeciągnij, by skopiować ją do końca tabeli.

	A	B	C	D	E	F
1	***x***	-2	-1	0	1	2
2	***y=2x***	=2*x				

Rys. 3. Uzupełniona tabela

Aby arkusz narysował wykres funkcji, trzeba zaznaczyć dane i wskazać właściwy typ wykresu. Do rysowania wykresu funkcji zawsze stosuje się wykres **Punktowy-XY** – wykres z połączonymi punktami jest najlepszy, ponieważ widać na nim cały przebieg funkcji.

- Zaznacz całą tabelę wraz z opisem argumentów i ich wartości.
- Na karcie **Wstawianie** w grupie **Wykresy** rozwiń listę, wybierz typ wykresu opisany jako **Punktowy-XY**, a następnie wybierz **Punktowy z prostymi liniami i znacznikami**.



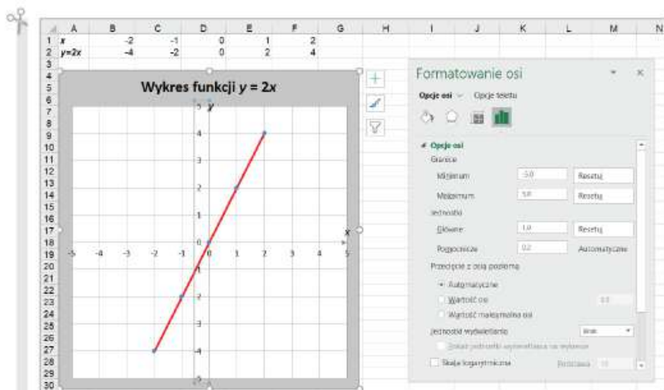
Rys. 4. Wstawianie wykresu punktowego

Wykres jest gotowy, ale trzeba nad nim jeszcze popracować – należy go opisać i sformatować jego elementy. Co warto poprawić?

- Tytuł wykresu – wystarczy kliknąć pole tekstowe **Tytuł wykresu** i wpisać odpowiedni tekst, np. **Wykres funkcji y = 2x**.
- Groty na osiach x i y – odpowiednie zakończenia linii można zmodyfikować po otwarciu menu podręcznego i kliknięciu **Formatuj oś** → **Wypełnienie i linia**.
- Tytuły osi poziomej (x) i pionowej (y) – można je wstawić na karcie **Projektowanie** → **Dodaj element wykresu** i przesunąć w odpowiednie miejsca przy osiach.
- Skalę przyjętą dla osi x i y – odpowiednie wartości **Minimum** i **Maksimum**, które ujednolicą skalę, można wybrać po dwukrotnym kliknięciu osi w oknie **Formatowanie osi**.
- Kształt wykresu – aby wykres zyskał kwadratowy kształt, należy odpowiednio rozciągnąć jego rogi.
- Kolor obszaru wykresu, kreślenia lub serii danych – edycję umożliwia dwukrotne kliknięcie dowolnego fragmentu wykresu.

Formatowanie wykresu za pomocą okna zadań Formatowanie

Poszczególne elementy wykresu można również formatować za pomocą okna zadań **Formatowanie**. Aby otworzyć okno, należy kliknąć wybrany element prawym przyciskiem myszy i wskazać polecenie **Formatuj <element wykresu>...**, np. po kliknięciu osi x należy wybrać **Formatuj oś...**, a następnie w oknie zadań **Formatowanie osi** wprowadzić odpowiednie zmiany.



Rys. 5. Ujednolicanie skali dla osi *x* i *y*

▶ Sformatuj wykres w najwygodniejszy dla ciebie sposób tak, aby był estetyczny i czytelny.

TRZY WYKRESY NARAZ

Umieść w jednym układzie współrzędnych wykresy trzech funkcji i sprawdź, czym się różnią: $y_1 = x + 5$; $y_2 = -2x + 3$; $y_3 = 4x$.

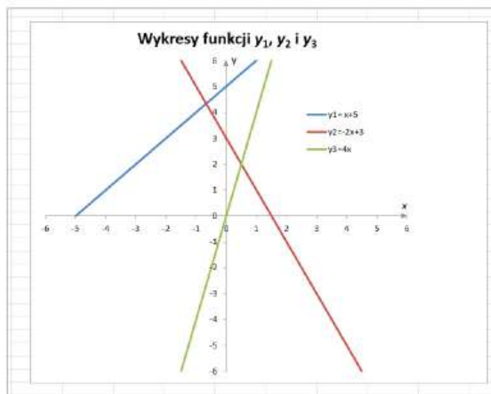
▶ Przejdź do kolejnego arkusza i przygotuj tabelę z danymi dla trzech funkcji równocześnie. Liczby do tabeli wybierz samodzielnie.

Q15

	A	B	C	D	E	F
1	x	-5	-4	0	4	5
2	$y_1 = x+5$	0	1	5	9	10
3	$y_2 = -2x+3$	13	11	3	-5	-7
4	$y_3 = 4x$	-20	-16	0	16	20
5						

Rys. 6. Tabela z przykładowymi danymi dla trzech funkcji

Po uzupełnieniu tabeli wstaw wykres na podstawie zawartych w niej danych. Może on wyglądać tak jak na rys. 7. Które z tych funkcji są rosnące, a które malejące? Które funkcje rosną lub maleją szybciej niż inne?



Rys. 7. Wykresy trzech funkcji liniowych w jednym układzie współrzędnych

WYKRES FUNKCJI MODUŁ.LICZBY (WARTOŚĆ BEZWZGLĘDNA)

A teraz coś trudniejszego – wykreśl funkcję $y = |x|$. Jak obliczyć wartości? Możesz oczywiście obliczyć moduł, czyli wartość bezwzględną każdego argumentu na papierze, a następnie wpisać do tabeli arkusza, jednak nie jest to konieczne. W Excelu masz do dyspozycji dużą bazę gotowych funkcji. Użyjesz jej do obliczeń.

- Przejdź do kolejnego arkusza i przygotuj tabelę z danymi dla funkcji $y = |x|$. Uzupełnij komórki dotyczące argumentów (np. –3, –2, –1, 0, 1 , 2, 3), a następnie ustaw kursor w komórce **B2** i kliknij przycisk **Wstaw funkcję (f_x)**. W oknie **Wstawianie funkcji** wybierz kategorię funkcji **Matematyczne**, a w niej – funkcję **MODUŁ.LICZBY**. Kliknij przycisk **OK**.

- W oknie **Argumenty funkcji** podaj dane do obliczeń – wskaż komórkę z argumentem, dla którego ma być obliczony moduł, a następnie skopiuj formułę do pozostałych komórek tabeli.

MODUŁ.LI... =MODUŁ.LICZBY(B1)

	A	B	C	D	E	F	G	H
1	x			-1	0	1	2	3
2	y=|x|	=MODUŁ.LICZBY(B1)		1	0	1	2	3
3		MODUŁ.LICZBY(liczba)						

Rys. 8. Uzupełnianie tabeli za pomocą funkcji **MODUŁ.LICZBY**

- Wykonaj wykres – w samej procedurze rysowania nic się nie zmienia.

ZADANIA

Podzielcie się na grupy. Wykonajcie zadania w jednym pliku, który w Dokumentach Google udostępni wam nauczyciel. Są w nim przygotowane zakładki dla poszczególnych grup i zadań. Opracujcie wykresy i odpowiedzcie na pytania postawione w zadaniach.

1. Umieśćcie na jednym rysunku cztery wykresy funkcji typu $y = ax$, różniące się tylko wartością współczynnika a. Jaki wpływ na wykres funkcji ma wartość tego współczynnika? Jakie wspólne cechy mają wszystkie wykresy?

2. Umieśćcie na jednym rysunku cztery wykresy funkcji typu $y = ax + b$, różniące się tylko wartością wyrazu wolnego b. Jaki wpływ na wykres funkcji ma wartość tego wyrazu? Jakie wspólne cechy mają wszystkie wykresy?

3. Umieśćcie na jednym rysunku wykresy następujących funkcji: $y_1 = |x|$; $y_2 = |x + 2|$; $y_3 = |x| + 2$; $y_4 = -|x|$; $y_5 = -|x + 2|$; $y_6 = -|x| + 2$. Jaką funkcję pełni wyraz wolny 2 w przypadku każdego z tych wykresów?

3.4 Przestawianie i przedstawianie danych

DOWIESZ SIĘ, JAK

- przeglądać i sortować duże zestawy danych w arkuszu kalkulacyjnym,
- tworzyć tabelę przestawną,
- korzystać z niektórych funkcji statystycznych.

Podczas tej lekcji poznasz narzędzia arkusza kalkulacyjnego umożliwiające sprawną pracę z dużym zestawem danych.

PRZEGLĄDANIE I SORTOWANIE ZESTAWÓW DANYCH

Materiał do pracy znajdziesz w pliku **oseski.xlsx**, który przekaże ci nauczyciel.

Zanim przystąpisz do analizy danych, na wszelki wypadek skopiuj zawartość arkusza **dane** – zaznacz cały arkusz (kliknij puste pole nad numerem pierwszego wiersza), skopiuj go i wklej do pierwszego pustego arkusza skoroszytu. Oryginalne dane warto zachować w niezmienionej postaci. Nowy arkusz nazwij **dane do obróbki**. Teraz możesz przystąpić do pracy.

Do arkusza możesz dopisać aktualne dane dotyczące urodzeń w twoim miejscu zamieszkania. W niektórych miastach i gminach spisy urodzeń są na bieżąco publikowane w sieci – sprawdź, czy są dostępne w twojej okolicy. Jeśli będziesz uzupełniać lub zmieniać dane, pamiętaj, aby:

- nic nie wpisywać, gdy brakuje dowolnej danej liczbowej (np. daty);
- wpisać znak * (lub inny), gdy brakuje danej tekstowej;
- datę urodzenia i godzinę wpisywać dokładnie według wzoru – w formacie daty;
- długość i wagę wpisywać jako liczby – bez jednostek.

► Dodaj kolumnę z numerami porządkowymi w arkuszu **dane do obróbki** – kliknij prawym przyciskiem myszy kolumnę **C** i z menu podręcznego wybierz **Wstaw**, a następnie **Cała kolumna**. Nazwij nową kolumnę (np. **Lp.**) i wypełnij ją serią danych, tak aby ponumerować noworodki. Przy okazji poznasz liczbę dzieci.

► W ten sam sposób dodaj kolumnę **F** zatytułowaną **Imię (pełne)**. Skopiuj do niej wszystkie pierwsze imiona, a potem przejrzyj je i zapisz w jednolitej postaci (np. Zosię zamień na Zofię, a Adasia na Adama).

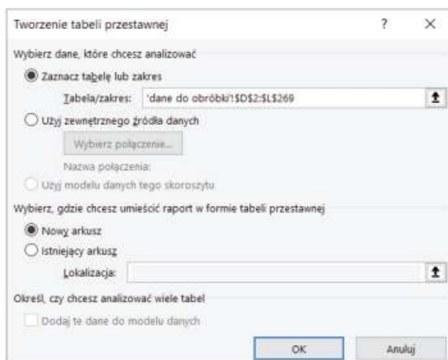
W jaki sposób szybko wyszukać w arkuszu dzieci o podanym imieniu, np. Anna? Należy posortować dane według imion – ustawić kursor w dowolnej komórce kolumny **Imię (pełne)** i wybrać na karcie **Narzędzia główne** przycisk **Sortuj i filtruj**.

- Wykorzystaj sortowanie i filtrowanie, aby sprawdzić, ile dzieci urodziło się w Szpitalu Bielańskim, ile dziewczynek przyszło na świat w Szpitalu MSWiA oraz ilu chłopców urodzonych w Szpitalu Matki i Dziecka ważyło co najmniej 3770 g.

TABELA PRZESTAWNA

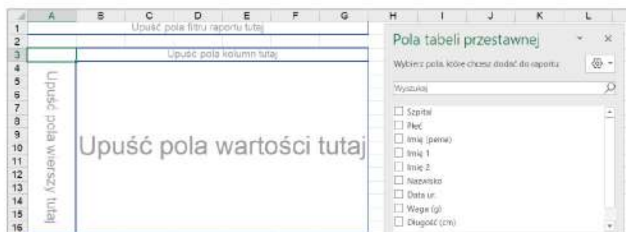
A jak opracować statystykę imion noworodków, czyli obliczyć, jak często występują poszczególne imiona? Do przestawiania oraz filtrowania kolumn i wierszy tak, by zaprezentować tylko niektóre dane z dużego zestawu, służy tabela przestawna.

- Ustaw kursor w obrębie danych, a następnie na karcie **Wstawianie** wybierz przycisk **Tabela przestawna**.
- W oknie **Tworzenie tabeli przestawnej** sprawdź, czy zakres danych został właściwie określony, zaznacz miejsce, w którym chcesz umieścić tabelę (na potrzeby tej lekcji wskaż **Nowy arkusz**), i kliknij przycisk **OK**.



Rys. 1. Tworzenie tabeli przestawnej

- Uzupełnij tabelę – skorzystaj z listy możliwych pól tabeli.
  - Przeciągnij nagłówki kolumn (nazwy pól) w odpowiednie miejsca tabeli: **Imię (pełne)** do miejsca oznaczonego **Upuść pola wierszy tutaj**, **Płeć** – do **Upuść pola kolumn tutaj** oraz ponownie **Imię (pełne)** – do miejsca oznaczonego **Upuść pola wartości tutaj**.



Rys. 2. Kreator tabeli przestawnej

Statystyka imion jest gotowa. Teraz wystarczy ustawić kursor w kolumnie **Suma końcowa** i posortować tabelę malejąco. Oto przykładowe wyniki sortowania tabeli.

	A	B	C	D
2				
3	Liczba z Imię (pełne)	Płeć		
4	Imię (pełne)	c	d	Suma końcowa
5	Wiktoria		8	8
6	Jan	8		8
7	Mateusz	7		7
8	Julia		7	7
9	Kacper	7		7
10	Natalia		6	6
11	Jakub	6		6
12	Karol	6		6
13	Maja		6	6
14	Zuzanna		6	6

Rys. 3. Przykładowa statystyka imion

Które imiona występują najczęściej w twoim zestawieniu?

FUNKCJA STATYSTYCZNA LICZ.JEŻELI

Czas na zmianę tematu – teraz zajmiesz się analizą ocen szkolnych. Nie będą to jednak stopnie liczbowe, ale literowe.

	A	B	C	D	E
1	**nazwisko**	**jęz. polski**	**matematyka**	**jęz. angielski**	**WF**
2	Adam Abacki	bdb	db	dst	dst
3	Bogdan Babacki	db	db	dop	bdb
4	Czesia Cabacka	dst	dst	dst	cel
5	Daniel Dabacki	db	bdb	db	db
6	Ewa Ebacka	dst	dop	ndst	bdb
7	Gabriel Gabacki	dst	cel	dst	dst
8	Krysia Kabacka	db	dst	bdb	db
9	Rafał Rabacki	ndst	dst	db	bdb

Rys. 4. Przykładowa tabela z ocenami literowymi

- Przygotuj własny arkusz według powyższego wzoru: dodaj inne przedmioty, wstaw imiona i nazwiska uczniów z twojej klasy oraz ich wymyślone literowe oceny z różnych przedmiotów.

Czy na ocenach w takiej postaci można cokolwiek obliczyć? Tak! Można np. przedstawić komplet wyników całej klasy.

F Do obliczeń należy użyć funkcji statystycznej **LICZ.JEŻELI**. Podczas wywoływania jej w każdej komórce, w której ma się pojawić wynik, trzeba podać obszar danych do przejrzenia i szukaną wartość.

- Pod tabelą z ocenami przygotuj tabelę do obliczeń i wyznacz liczbę poszczególnych ocen z danego przedmiotu – zastosuj funkcję **LICZ.JEŻELI**.

- Wyniki obliczone przez funkcję zsumuj w kolumnie **RAZEM**.

12						
13						**RAZEM**
14	**cel**					
15	**bdb**					
16	**db**					
17	**dst**					
18	**dop**					
19	**ndst**					
20						
21						

Rys. 5. Tabela do obliczeń

- W osobnym arkuszu wykonaj wykres kołowy ocen klasowych, zawierający nazwy stopni i ich liczbę – wybierz diagram trójwymiarowy (**Kołowy 3-W**), wyłącz legendę, dodaj etykiety i procenty.

ZADANIA

1. Posłuż się funkcją **LICZ.JEŻELI**, aby obliczyć, ile jest dziewczynek, a ilu chłopców w analizowanym zestawie danych o noworodkach.

2. W arkuszu **dane do obróbki** wypisz wszystkie wartości długości ciała noworodków (od najmniejszej do największej). Oblicz, ile jest noworodków o danej długości ciała, a następnie przedstaw wyniki na wykresie kolumnowym.

3.5 Dużo danych

DOWIESZ SIĘ, JAK

- przeglądać i analizować w arkuszu duży zestaw danych,
- korzystać z niektórych funkcji statystycznych,
- uczestniczyć w projektach przetwarzania rozproszonego.

Gdy danych jest naprawdę dużo, wówczas do ich obróbki przydaje się statystyka, ale gdy danych jest za dużo... wtedy żaden komputer sam nie da rady ich przetworzyć. W takiej sytuacji należy podzielić zadanie na wiele mniejszych i zaprosić do realizacji projektu jak najwięcej uczestników. Przekonasz się o tym podczas tej lekcji.

FUNKCJE STATYSTYCZNE

Do pracy będziesz potrzebować pliku z poprzedniej lekcji – **oseski.xlsx**.

F

► Otwórz opracowany na poprzedniej lekcji plik i wyznacz w arkuszu średnią arytmetyczną masy i długości ciała noworodków. Umieść kursor w komórce przeznaczonej na wynik, kliknij przycisk f_x i w oknie **Wstawianie funkcji** wybierz kategorię funkcji **Statystyczne**, a w niej funkcję **ŚREDNIA**. Na koniec zaznacz odpowiedni zakres danych.

► Do przydatnych funkcji statystycznych należą też **MIN**, **MAX** i **MEDIANA**, które służą do znajdowania odpowiednio: najmniejszej, największej i środkowej wartości z danych. Zastosuj je w swoim arkuszu, by obliczyć wartości minimalne, środkowe i maksymalne wagi i wzrostu noworodków.

=ŚREDNIA(K2:K268)	=ŚREDNIA(L2:L268)	← wartość średnia podanego zakresu danych
=MIN(K2:K268)	=MIN(L2:L268)	← wartość najmniejsza
=MEDIANA(K2:K268)	=MEDIANA(L2:L268)	← wartość środkowa
=MAX(K2:K268)	=MAX(L2:L268)	← wartość największa

Rys. 1. Formuły **ŚREDNIA**, **MIN**, **MEDIANA**, **MAX** z przykładowym zakresem danych

W przypadku średniej, minimum i maksimum funkcję można również wybrać z listy przycisku Σ **Autosumowanie**.

D	E	F	G	H
19.05.2017	03:40	3100	53	Gdańsk
19.05.2017	02:55	3550	57	Gdańsk
19.05.2017	03:38	3330	54	Gdańsk
17.05.2017	02:25	3500	57	Gdańsk
17.05.2017	16:57	3800	56	Gdańsk
	średnia	**3446,46**	**55,09**	
	minimum	1700	43	
	mediana	3470	55	
	maksimum	5040	63	

Rys. 2. Przykładowe wyniki obliczeń – przeciętny noworodek waży ok. 3,5 kg i ma ok. 55 cm długości

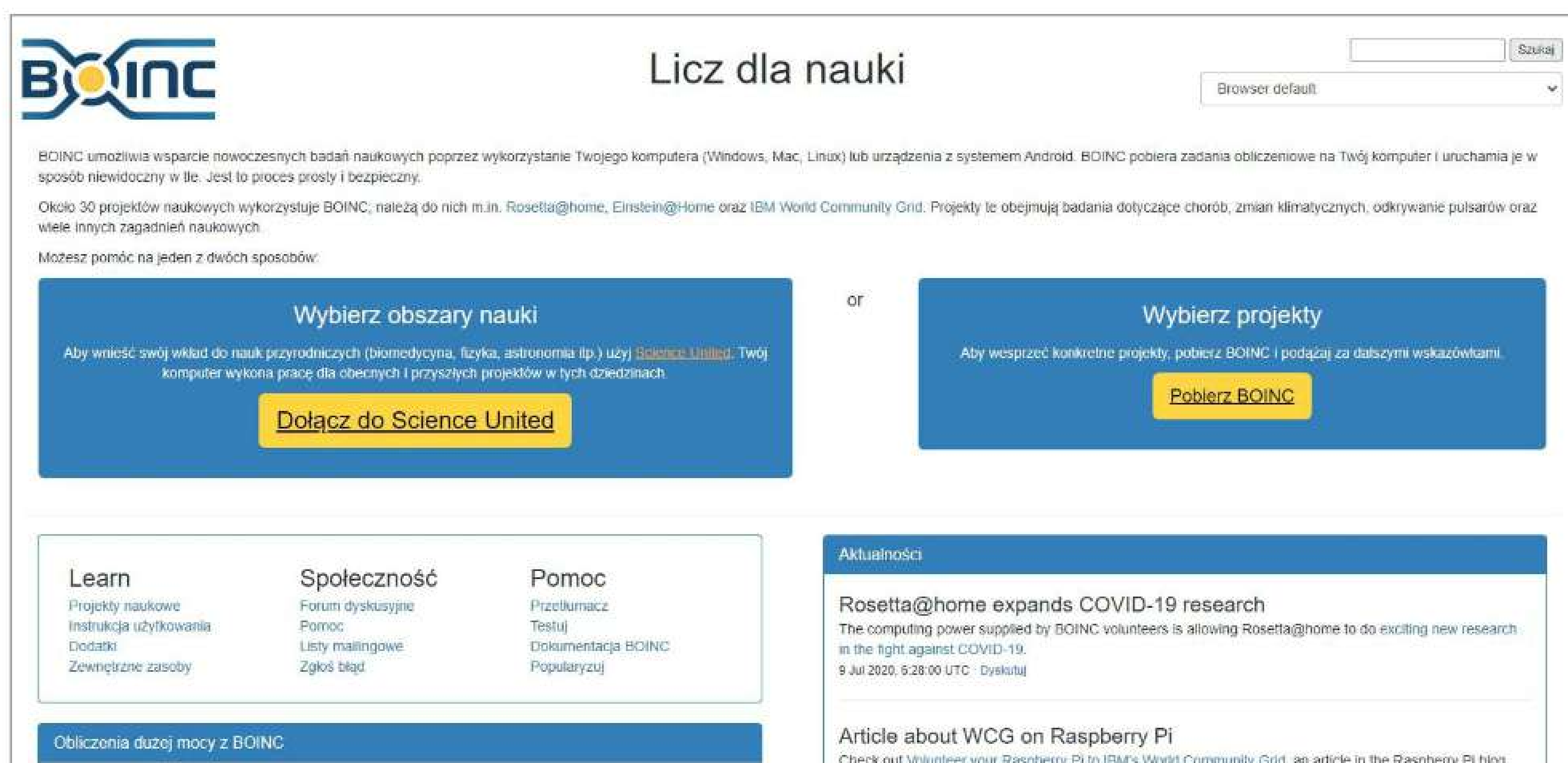
GDY DANYCH JEST ZA DUŻO...

W przypadku projektów, w których występuje naprawdę dużo danych, gdy już żaden, nawet najsilniejszy komputer nie jest w stanie ich samodzielnie przetworzyć, stosuje się przetwarzanie rozproszone. Organizatorzy projektu przygotowują oprogramowanie do obliczeń i udostępniają je w sieci uczestnikom projektu. Zazwyczaj działanie takiego programu odbywa się w tle innych czynności wykonywanych przez komputer lub wtedy, gdy użytkownik nie pracuje, a komputer pozostawia włączony.

Właściciel komputera rejestruje się jako użytkownik projektu. Oprogramowanie zainstalowane na jego komputerze łączy się z serwerem projektu i pobiera do opracowania porcję danych. Po wykonaniu obliczeń przesyła wyniki do centrum, pobiera następną porcję i tak dalej. Wyniki uczestnika projektu są sprawdzane, wysyłane do dalszej obróbki i odnotowywane. Może on na stronach projektu sprawdzić, ile danych już opracował i które miejsce zajmuje w rankingu wszystkich uczestników. Może też z nimi podyskutować na forum poświęconym projektowi. Wyniki i kolejne etapy obliczeń są publikowane na stronach projektu, aby każdy z uczestników wiedział, na jakim etapie jest problem, w którego rozwiązaniu uczestniczy.

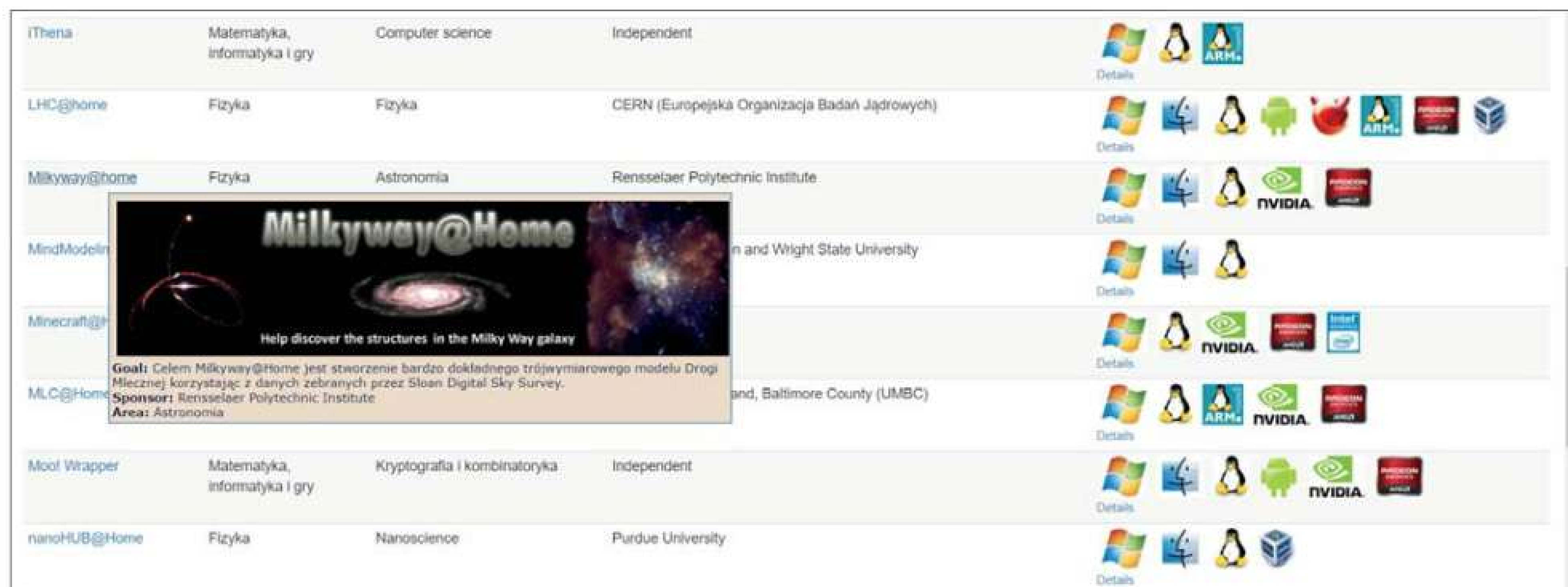
W internecie powstaje wiele platform, na których można realizować różne projekty. Wspólna platforma oprogramowania łączy projekty o podobnej tematyce i umożliwia uczestnikowi udział w kilku projektach równocześnie. Lista projektów przetwarzania rozproszonego działających na różnym oprogramowaniu znajduje się pod adresem **distributedcomputing.info/projects.html**.

Od kilku lat największą popularnością cieszy się platforma BOINC (akronim od ang. *Berkeley Open Infrastructure for Network Computing*; czytaj: berkli ołpen infrastrakczer for netłork kompjutin; otwarta infrastruktura przetwarzania rozproszonego Berkeley), która działa pod adresem **boinc.berkeley.edu**.



Rys. 3. Fragment strony głównej platformy BOINC

Część materiałów na stronie BOINC została spolszczona. Inne szczegółowe informacje na temat przetwarzania rozproszonego i realizowanych w ten sposób projektów znajdziesz w polskojęzycznym serwisie **www.boincatpoland.org**. Dowiesz się tam także, w jaki sposób możesz przystąpić do któregoś z nich. Lista projektów BOINC jest dostępna pod adresem: **https://boinc.berkeley.edu/projects.php**.



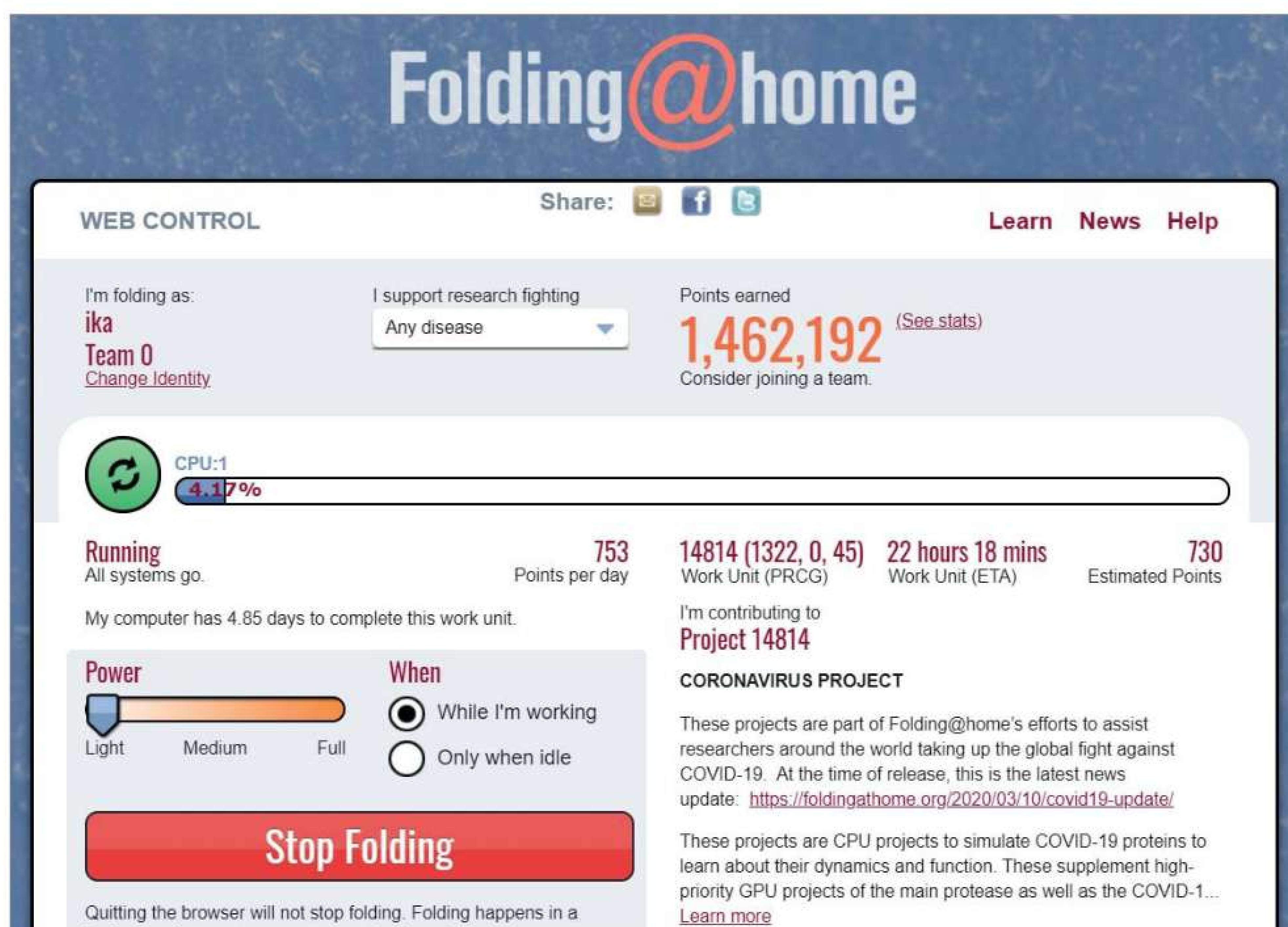
Rys. 4. Fragment listy projektów – po najechaniu wskaźnikiem myszy na nazwę projektu otwiera się okno z opisem celu projektu

Ulubionym projektem autorów tego podręcznika jest **SETI@home** (**setiathome.berkeley.edu**). SETI to akronim Search for Extraterrestrial Intelligence (czytaj: sercz for ekstraterestjal intelidżens) – poszukiwanie inteligencji pozaziemskiej. SETI prowadzi nasłuch wąskiego pasma sygnałów radiowych z kosmosu z wykorzystaniem radioteleskopów. Takie sygnały nie występują naturalnie, ich wykrycie powinno więc dostarczyć dowodów na istnienie pozaziemskiej technologii. Projekt niedawno po ponad 20 latach zawiesił działalność w formule przetwarzania rozproszonego – teraz naukowcy będą analizować ogromną ilość przetworzonych i zgromadzonych danych.

Ludzie pracujący przy takich projektach nie skupiają się tylko na specjalistycznych badaniach naukowych – reagują błyskawicznie na to, co dzieje się na świecie. Gdy dotknęła nas epidemia COVID-19, w ramach projektów zajmujących się poszukiwaniem leków i badaniem genomu natychmiast rozpoczęły się nowe badania. Leku szukają badacze zaangażowani w projekt platformy BOINC **Rosetta@home**. Po wybuchu epidemii został on wykorzystany do przewidywania struktury białek ważnych dla choroby, a także do wytworzenia nowych, stabilnych minibiałek do wykorzystania w charakterze potencjalnych środków terapeutycznych i diagnostycznych, wiążących się z białkiem SARS-CoV-2.

Rys. 5. Przetwarzanie danych projektu Rosetta@home na platformie BOINC – badanie białek wirusa COVID-19

Inny projekt, w ramach którego prowadzi się walkę z wirusem, to **Folding@home**, działający na własnej platformie. Jest on poświęcony chorobom wynikającym z nieprawidłowego fałdowania i agregacji białek oraz nowym, obliczeniowym sposobom opracowywania leków.



Rys. 6. Przetwarzanie danych dotyczących choroby koronawirusowej COVID-19

Zarządzana przez IBM, samodzielna niegdyś platforma World Community Grid (**worldcommunitygrid.org**) dziś jest zintegrowana z BOINC. Związane z nią osoby koncentrują się na przedsięwzięciach mających na celu niesienie pomocy ludzkości. Na stronach serwisu można obejrzeć długą listę już ukończonych projektów. Projekty aktywne w 2020 r. to:

- **OpenPandemics – COVID-19** (czytaj: ołpenpandemiks kołwid najntin) – wyszukiwanie potencjalnych metod leczenia;
- **Africa Rainfall Project** (czytaj: afrika rejnfol prodżekt) – prognozowanie opadów w Afryce;
- **Microbiome Immunity Project** (czytaj: majkrobajom imjuniti prodżekt) – kompleksowe badanie ludzkiego mikrobiomu;
- **Smash Childhood Cancer** (czytaj: smasz czajldhud kanser) – walka z rakiem u dzieci;
- **Help Stop TB** (czytaj: help stop tibi) – poszukiwanie leku na antybiotykooporną (wielolekooporną) gruźlicę;
- **Mapping Cancer Markers** (czytaj: maping kanser markers) – identyfikacja markerów (znaczników) związanych z różnymi typami nowotworów.

Projekt **LHC@home** (**lhcathome.web.cern.ch**) wspiera działanie Wielkiego Zderzacza Hadronów – największego na świecie urządzenia służącego do przyspieszania cząstek (hadronów) do prędkości bliskich prędkości światła w próżni. Akcelerator ten znajduje

się na głębokości 100 m pod ziemią w laboratorium Europejskiego Ośrodka Badań Jądrowych CERN pod Genewą. Dwa strumienie protonów są rozpędzane do prędkości bliskiej prędkości światła, a następnie doprowadzane do „czołowego zderzenia". Ilość danych uzyskana z każdego eksperymentu stanowi niemałe wyzwanie pod względem obliczeń i przechowywania. Zderzenia protonów następują 30 milionów razy w ciągu sekundy, a detektory Wielkiego Zderzacza Hadronów produkują około 100 TB danych na sekundę. Uruchomiony w 2009 r. projekt LHC@home za pomocą wirtualizacji stara się dobrać optymalne parametry torów cząsteczek w celu osiągnięcia jak najlepszej stabilności rozpędzanych cząstek. Rzecz w tym, aby wyprowadzić cząstki na stabilną kołową orbitę, na której utrzymają się przez kilkaset tysięcy okrążeń akceleratora o długości 27 km.



Rys. 7. Fragment strony głównej projektu LHC@home

Z kolei **Universe@home** (**universeathome.pl**) jest projektem utworzonym na Uniwersytecie Warszawskim. Osoby zaangażowane w ten projekt zajmują się opracowaniem symulacji powstawania obiektów kosmicznych (głównie, ale nie tylko, gwiazdowych).

Zanim dołączysz...

Zanim dołączysz do projektu, uzyskaj na to zgodę rodziców lub opiekunów (gdy korzystasz z komputera w domu) lub nauczyciela (gdy dotyczy to komputerów szkolnych).

ZADANIA

F

1. Oblicz średnie, maksymalne i minimalne wagi i długości ciała noworodków w podziale na płeć. Aby prawidłowo wybrać dane, posortuj je, przefiltruj lub umieść w tabeli przestawnej.
2. Znajdź na platformie BOINC dwa lub trzy projekty, które najbardziej ci się podobają. Sporządź notatkę na temat korzyści wynikających z ich realizacji.

3.6 Moi znajomi

DOWIESZ SIĘ,

- co to jest kartotekowa baza danych,
- jak filtrować i sortować dane w bazie,
- jak zastosować formularz do wpisywania danych do bazy.

Podczas tej lekcji utworzysz w arkuszu kalkulacyjnym w chmurze kartotekową bazę danych, zawierającą informacje o czytelnictwie uczniów uczęszczających do twojej klasy.

CZYM JEST KARTOTEKOWA BAZA DANYCH

Komputerowa baza danych to zbiór danych cyfrowych, które zapisano zgodnie z pewnymi określonymi regułami, obowiązującymi w programie przeznaczonym do zarządzania bazą danych.

Kartotekowa baza danych to najprostsza baza – złożona z jednej tabeli, zbudowanej z komórek o identycznej strukturze. Tego typu bazy danych wykorzystuje się do tworzenia wszelkiego rodzaju spisów, np. książek lub danych osobowych. Można je utworzyć zarówno w programie zarządzającym bazami danych (np. Access z pakietu Office, Base z pakietów Apache OpenOffice i LibreOffice), jak i w arkuszach kalkulacyjnych czy edytorach tekstów.

Rys. 1. Przykładowa kartotekowa baza danych – spis kontaktów w telefonie

PRZYGOTOWANIE DANYCH

Każdą bazę trzeba odpowiednio zaprojektować. Baza czytelnictwa może przyjąć postać prostej tabeli, takiej jak na rys. 2. Zestaw danych dotyczących jednej osoby (cały wiersz) to jeden **rekord**.

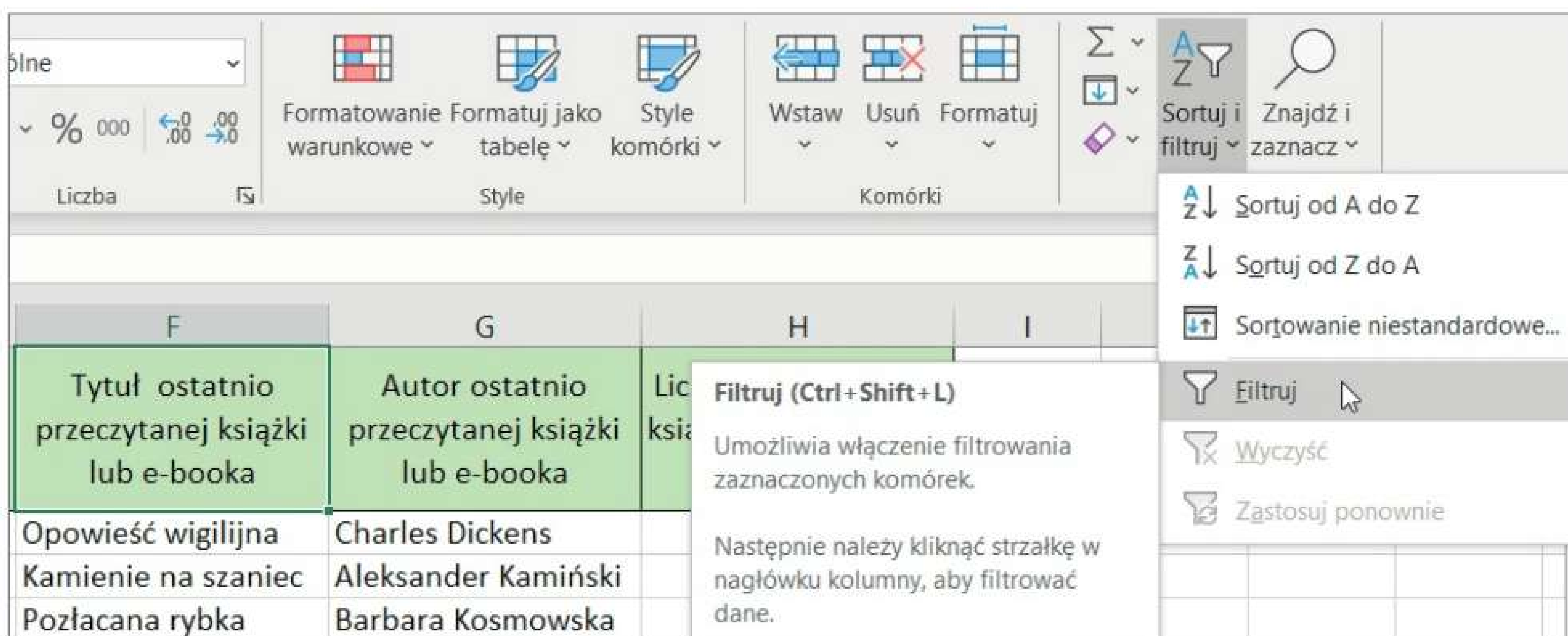
	A	B	C	D	E	F	G	H
1	L.p.	Imię	Nazwisko	Liczba przeczytanych książek w ciągu ostatnich 3 miesięcy	Liczba przeczytanych e-booków w ciągu ostatnich 3 miesięcy	Tytuł ostatnio przeczytanej książki lub e-booka	Autor ostatnio przeczytanej książki lub e-booka	Liczba przeczytanych książek i e-booków w ciągu roku
2	1							
3	2							
4	3							

Rys. 2. Projekt tabeli

► Uzupełnij tabelę udostępnioną przez nauczyciela za pośrednictwem Dysku Google.

FILTROWANIE DANYCH

Baza utworzona w arkuszu umożliwia szybkie przeszukiwanie danych. Aby uruchomić filtrowanie, należy kliknąć myszą w obszarze danych, a następnie na karcie **Narzędzia główne** w grupie **Edytowanie** wybrać listę **Sortuj i filtruj** → **Filtruj**.



Rys. 3. Włączanie filtrowania

Po uruchomieniu funkcji filtrowania w prawym dolnym rogu każdego pola w nagłówku tabeli pojawi się mały kwadratowy przycisk ze strzałką.

	A	B	C	D	E	F	G	H
1	L.p.	Imię	Nazwisko	Liczba przeczytanych książek w ciągu ostatnich 3 miesięc	Liczba przeczytanych e-booków w ciągu ostatnich 3 miesięc	Tytuł ostatnio przeczytanej książki lub e-booka	Autor ostatnio przeczytanej książki lub e-booka	Liczba przeczytanych książek i e-booków w ciągu roku
2	1	Adam	Abacki	2	0	Opowieść wigilijna	Charles Dickens	20
3	2	Helenka	Dabacka	3	1	Kamienie na szaniec	Aleksander Kamiński	15
4	3	Paweł	Gabacki	0	4	Pozłacana rybka	Barbara Kosmowska	8
5	4	Tosia	Kowalska	0	2	Pozłacana rybka	Barbara Kosmowska	11
6	5	Antoni	Lewandowski	1	2	Artysta	Sławomir Mrożek	20
7	6	Nina	Nowak	2	1	A.B.C.	Agatha Christie	5

Rys. 4. Arkusz z włączonym filtrowaniem danych

Aby ograniczyć wyświetlanie rekordów do tych, które spełniają pewne kryterium, trzeba założyć filtr – kliknąć strzałkę i wybrać filtr, który ma być zastosowany w danej kolumnie. Jeśli chcesz się dowiedzieć, które osoby ostatnio przeczytały *Pozłacaną rybkę*, musisz kliknąć strzałkę w polu **Tytuł ostatnio przeczytanej książki lub e-booka** i jako kryterium wybrać *Pozłacaną rybkę*.

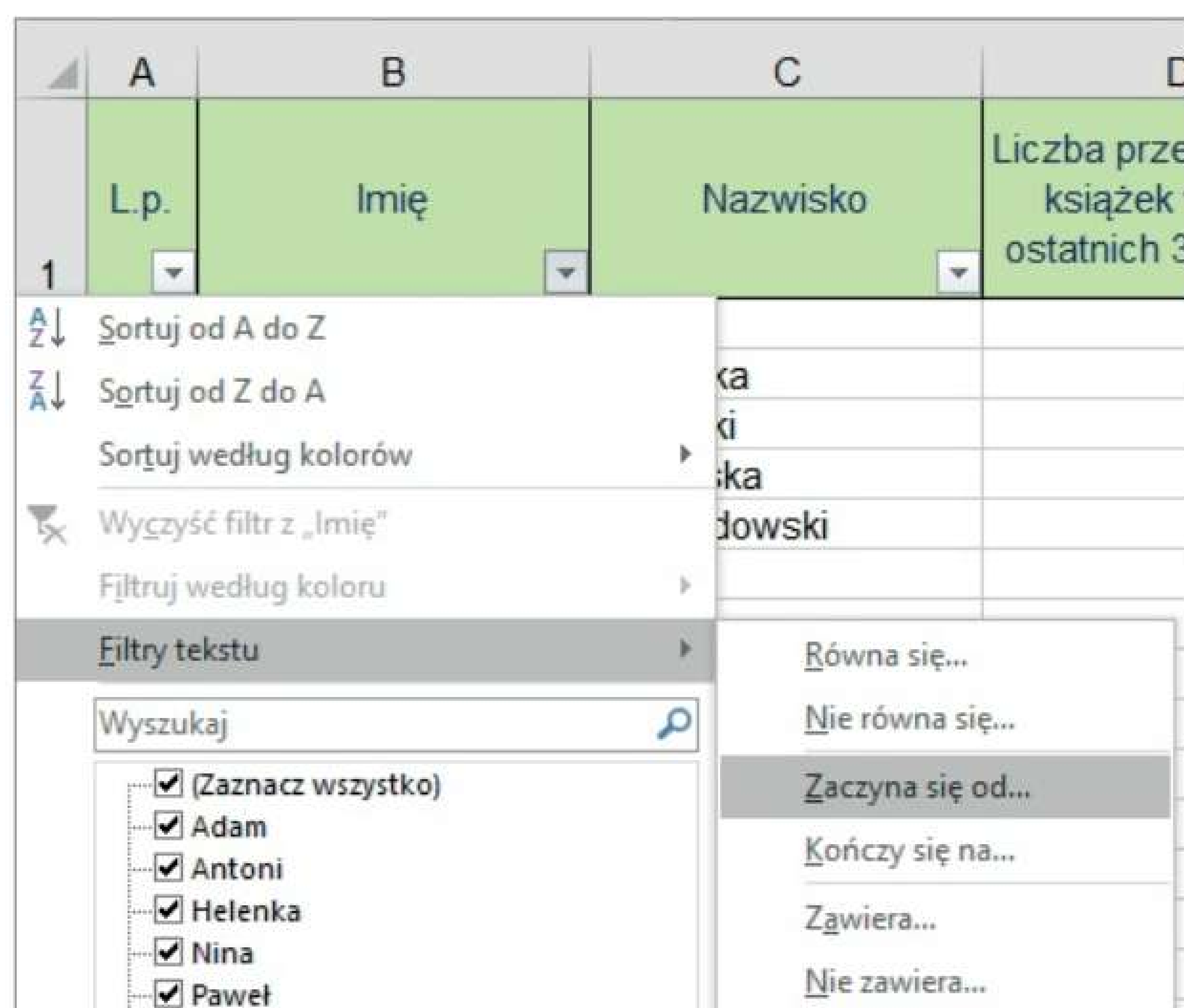
	A	B	C	D	E	F	G	H
1	L.p.	Imię	Nazwisko	Liczba przeczytanych książek w ciągu ostatnich 3 miesięc	Liczba przeczytanych e-booków w ciągu ostatnich 3 miesięc	Tytuł ostatnio przeczytanej książki lub e-booka	Autor ostatnio przeczytanej książki lub e-booka	Liczba przeczytanych książek i e-booków w ciągu roku
4	3	Paweł	Gabacki	0	4	Pozłacana rybka	Barbara Kosmowska	8
5	4	Tosia	Kowalska	0	2	Pozłacana rybka	Barbara Kosmowska	11

Rys. 5. Przefiltrowane dane osób

Zwróć uwagę na rysunek lejka obok pomniejszonej strzałki w wybranym przed chwilą polu oraz na numery wierszy, których dotyczy wybrane kryterium. Pozostałe wiersze są ukryte.

Aby usunąć założony filtr i ponownie wyświetlić wszystkie rekordy, wystarczy kliknąć strzałkę, wybrać opcję **Zaznacz wszystko** i zatwierdzić ją przyciskiem **OK**.

Kryteria filtrowania można ustawiać samodzielnie. W kolumnach z danymi liczbowymi po kliknięciu strzałki przy wybranym nagłówku należy rozwinąć w prawo **Filtry liczb**, które pozwalają np. wyświetlić tylko liczby z danego zakresu. W kolumnach z danymi tekstowymi lub tekstowo-liczbowymi po kliknięciu strzałki przy wybranym nagłówku należy rozwinąć w prawo **Filtry tekstu**, które pozwalają np. wyświetlić tylko wyniki dla nazwisk zaczynających się od wskazanej litery.



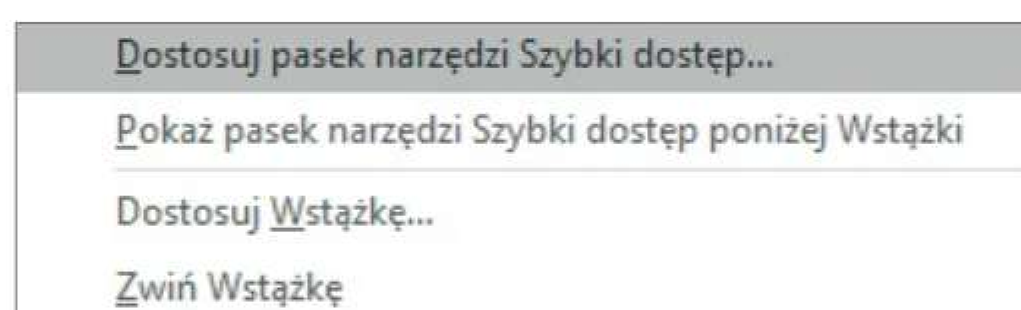
Rys. 6. Wybór filtrów tekstu

▶ Zastosuj filtr tekstu, aby wyświetlić tylko te rekordy, w których imię zaczyna się na literę A.

UZUPEŁNIANIE I POPRAWIANIE DANYCH

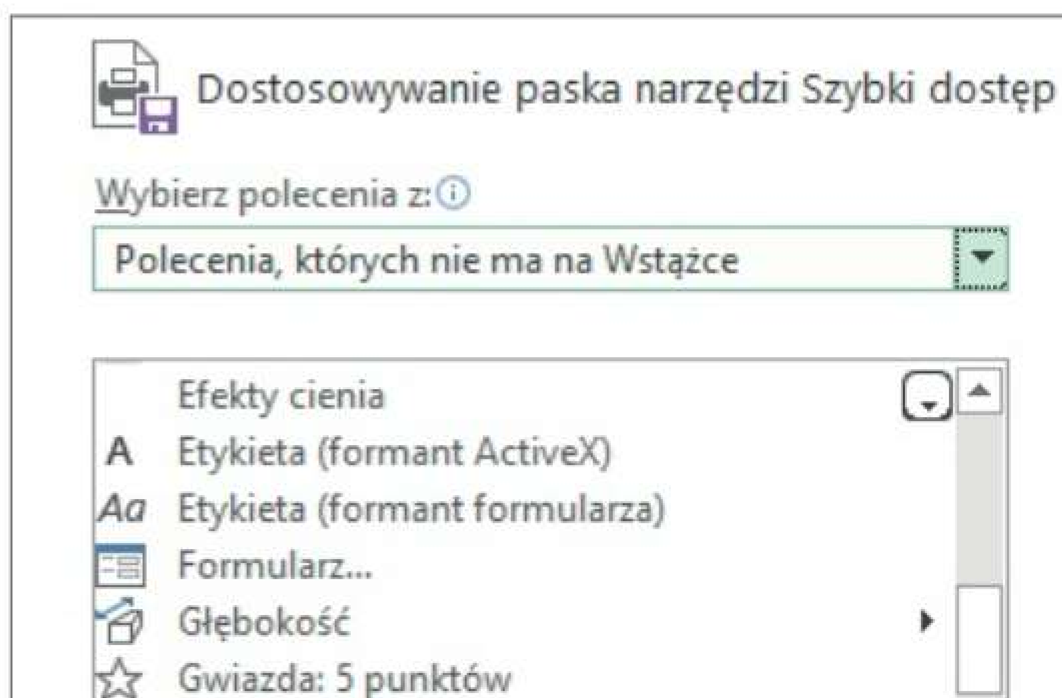
Pojedyncze rekordy najłatwiej dopisywać lub poprawiać za pomocą **Formularza** – w ten sposób cały rekord jest dobrze widoczny, a w arkuszu nie zawsze mieści się on na ekranie. Jeśli nie widzisz przycisku do uruchamiania **Formularza** na pasku narzędzi, musisz go dodać.

▶ Kliknij prawym przyciskiem myszy z prawej strony paska narzędzi i wybierz opcję **Dostosuj pasek narzędzi Szybki dostęp...**, aby otworzyć okno **Opcje programu Excel**.



Rys. 7. Wybieranie opcji **Dostosuj pasek narzędzi Szybki dostęp...**

- Jeśli wśród wyświetlonych automatycznie popularnych poleceń nie znajdziesz przycisku **Formularza**, wybierz z rozwijanej listy u góry okna **Polecenia, których nie ma na Wstążce**, znajdź przycisk, kliknij go, aby podświetlił się na niebiesko, a następnie kliknij **Dodaj >> OK**. Przycisk pojawi się na pasku narzędzi **Szybki dostęp**.



Rys. 8. Dostosowywanie paska narzędzi

- Kliknij dodany przycisk i w wyświetlonym oknie wpisz dane nowej osoby.
- Aby poprawić dane, przemieszczaj się między kolejnymi rekordami za pomocą przycisków **Znajdź poprzedni** i **Znajdź następny**.

znajomi ? ×

L.p.:
Imię:
Nazwisko:
Liczba przeczytanych książek w ciągu ostatnich 3 miesięcy:
Liczba przeczytanych e-booków w ciągu ostatnich 3 miesięcy:
Tytuł ostatnio przeczytanej książki lub e-booka:
Autor ostatnio przeczytanej książki lub e-booka:
Liczba przeczytanych książek i e-booków w ciągu roku:

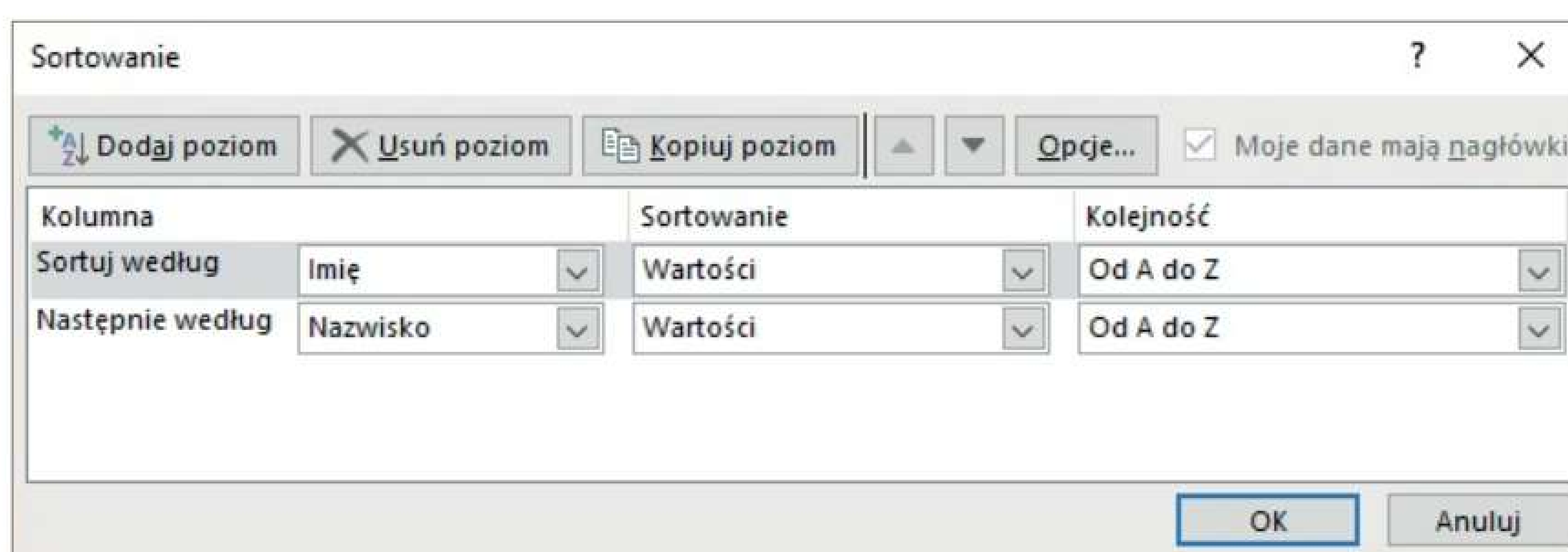
Nowy rekord
Nowy
Usuń
Przywróć
Znajdź poprz
Znajdź następ
Kryteria
Zamknij

Rys. 9. Edycja danych w formularzu

SORTOWANIE DANYCH

Na pewno pamiętasz, jak uporządkować dane za pomocą przycisku **Sortuj od A do Z**. Teraz dowiesz się, w jaki sposób wykonać sortowanie niestandardowe, np. posortować listę alfabetycznie według imion i nazwisk (w przypadku dwóch takich samych imion).

- Zaznacz obszar danych i na karcie **Narzędzia główne** w grupie **Edytowanie** wybierz polecenie **Sortuj i filtruj** → **Sortowanie niestandardowe**. Otworzy się okno **Sortowanie**.
- Zwróć uwagę, czy jest zaznaczone pole **Moje dane mają nagłówki**, aby tytuły kolumn pozostały na swoim miejscu.
- W polu **Sortuj według** wybierz pole, według którego chcesz posortować bazę (**Imię**).
- Dodaj następny poziom i wybierz kolejne pole, według którego chcesz sortować (**Nazwisko**).



Rys. 10. Sortowanie niestandardowe

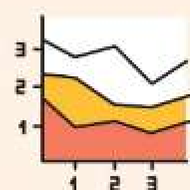
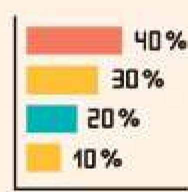
ZADANIA

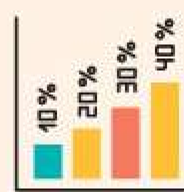
1. Sprawdź, kto z twoich koleżanek i kolegów przeczytał najwięcej książek i e-booków w ciągu roku. Jaką metodą można to zrobić najprościej?
2. Oblicz, ile osób z twojej klasy przeczytało wybraną książkę. Skorzystaj z funkcji **LICZ.JEŻELI**.

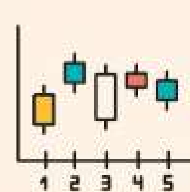
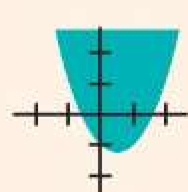
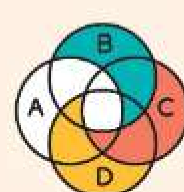
Jak usprawnić pracę z arkuszem kalkulacyjnym?

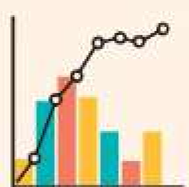
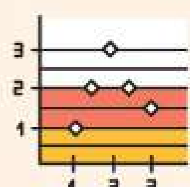
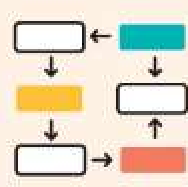
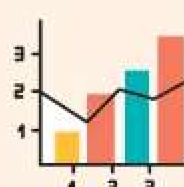
Korzystaj z wbudowanych narzędzi arkusza, skrótów klawiaturowych i metod szybkiego zaznaczania. Sortuj i filtruj dane, używaj tabel przestawnych. Używaj gotowych funkcji, zamiast wprowadzać własne formuły. Stosuj adresy względne i bezwzględne dla wygody kopiowania danych i czytelności formuł. Rozdzielaj dane między kolejne arkusze.

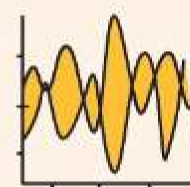
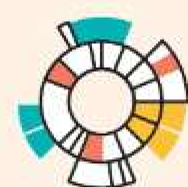
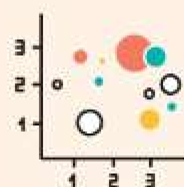




Jak konstruować wykresy?

Dobieraj typ wykresu do ilustrowanych danych (wykres punktowy XY dla funkcji liniowej, wykres kołowy dla danych procentowych). Używaj narzędzi graficznych do wprowadzania szybkich zmian. Stosuj wbudowane metody formatowania wykresów.

Jak wykorzystać optymalnie moc swojego komputera?

Weź udział w jednym lub kilku wybranych projektach przetwarzania rozproszonego – niech komputer pracuje, gdy śpisz lub odpoczywasz. Wybierz te projekty, które cię interesują, lub te, które uważasz za najbardziej pożyteczne. Małe cząstki działań uczestników mogą dać efekt ważny dla całej ludzkości.



4 Lekcje z modelami



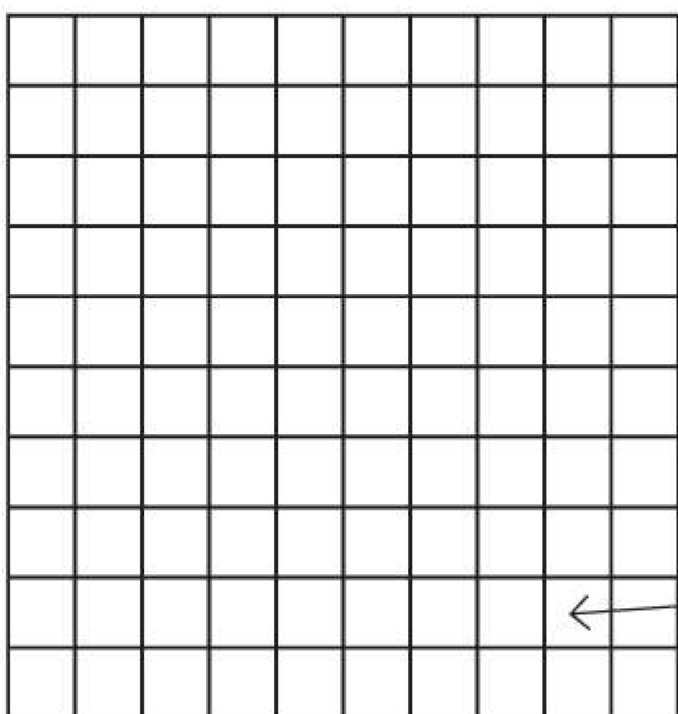


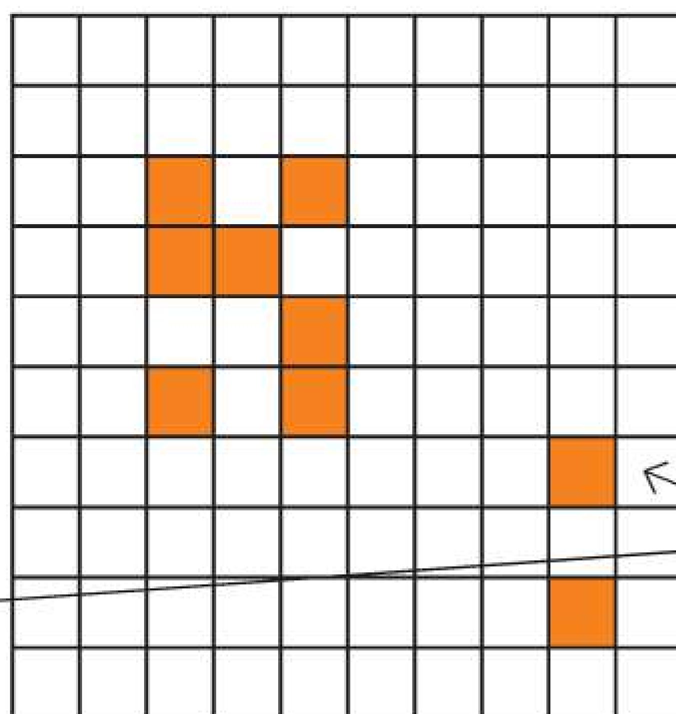
Model to nie tylko osoba zawodowo pozująca artystom lub prezentująca stroje. Model to również odwzorowanie w pewnej skali większego obiektu, np. samolotu albo budynku, reprezentacja świata w umyśle człowieka (nie zawsze zgodna z rzeczywistością), a także przedmiot badań – podobizna oryginału, której badanie pozwala otrzymywać informacje na temat rzeczywistości.

GRA W ŻYCIE

PLANETA I JEJ MIESZKAŃCY

Planetę – złożoną z kwadracików tworzących dwuwymiarową siatkę – zamieszkują pomarańczowe komórki.



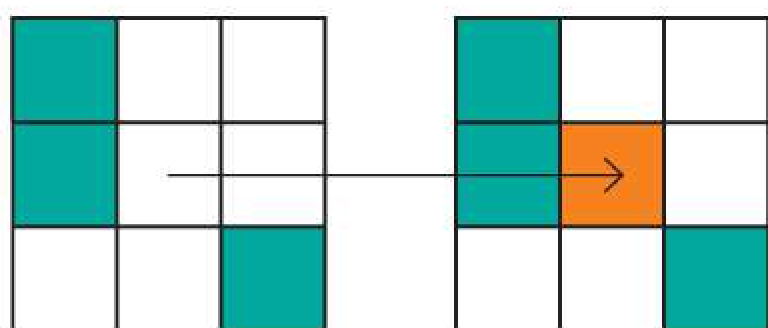


Białe kwadraciki to miejsca puste, które zostały zwolnione po śmierci komórki żywej lub nigdy nie były zajęte.

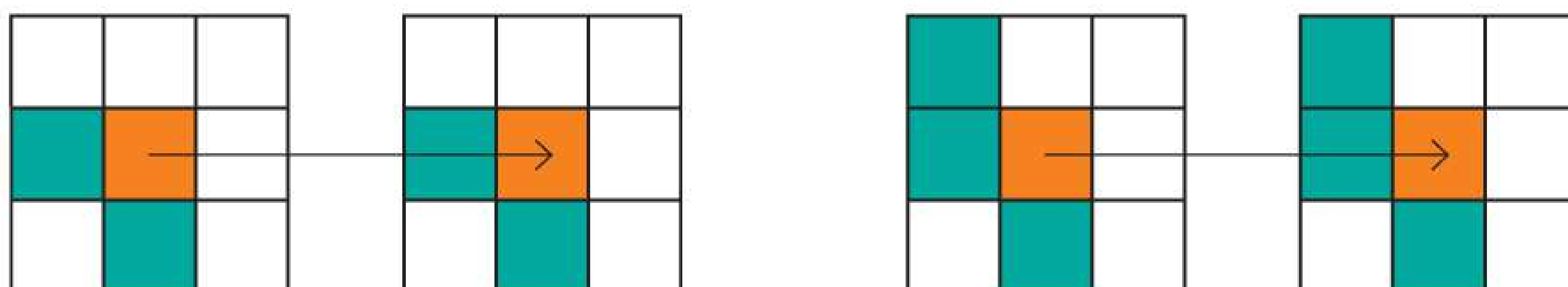
REGUŁY GRY

Gra w życie jest specyficzna – po wprowadzeniu danych wejściowych nie ingeruje się w jej przebieg. Stan populacji zależy wyłącznie od parametrów początkowych oraz ustalonych reguł. Status wszystkich komórek zmienia się w określonych jednostkach czasu dokładnie w tym samym momencie. Stan komórki zależy od liczby jej żywych sąsiadów bezpośrednio przylegających do niej ścianami lub rogami.

Komórka pusta ożywa, gdy zostanie otoczona przez trzech żywych sąsiadów (zielone komórki), np.:

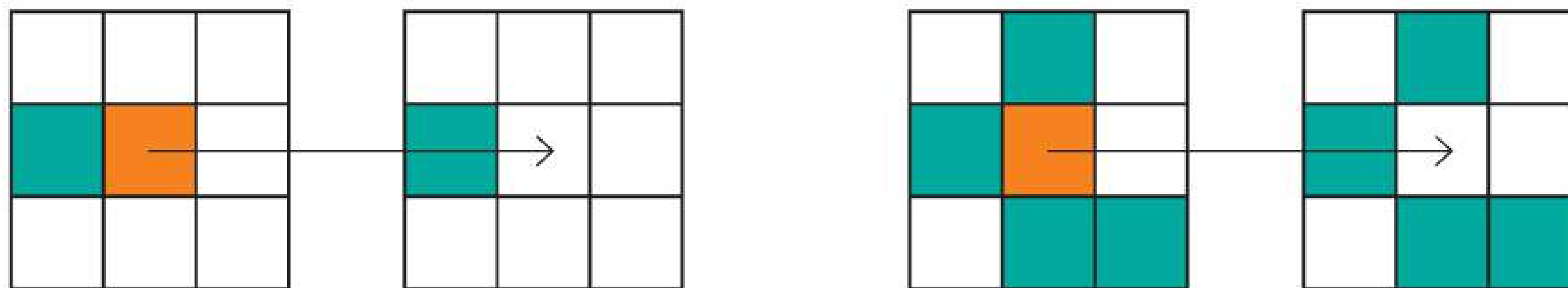


Komórka żywa pozostaje przy życiu, gdy ma dwóch lub trzech sąsiadów, np.:

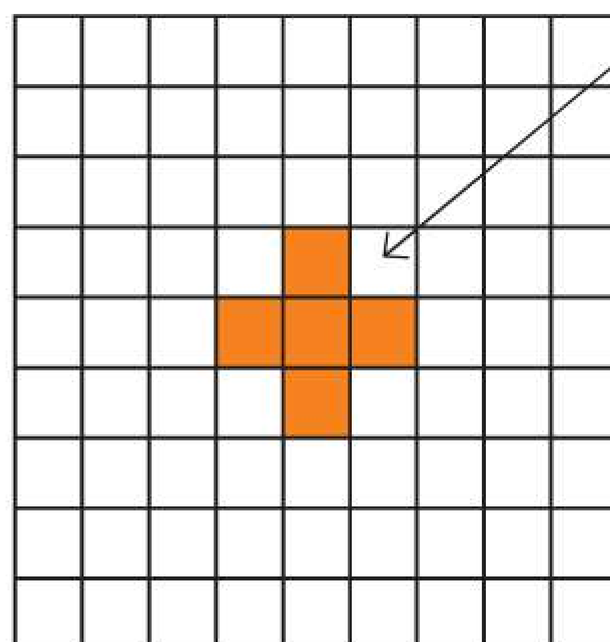


Czy da się przewidzieć, jak będzie się rozwijało życie na naszej planecie? Problem ten roztrząsało wielu filozofów, myślicieli i naukowców. Pod koniec lat 60. XX w. zainteresował on też brytyjskiego matematyka Johna Conwaya (czytaj: dżona konłeja), który ostatecznie w 1970 r. stworzył *Grę w życie* – symulator narodzin, życia i śmierci populacji.

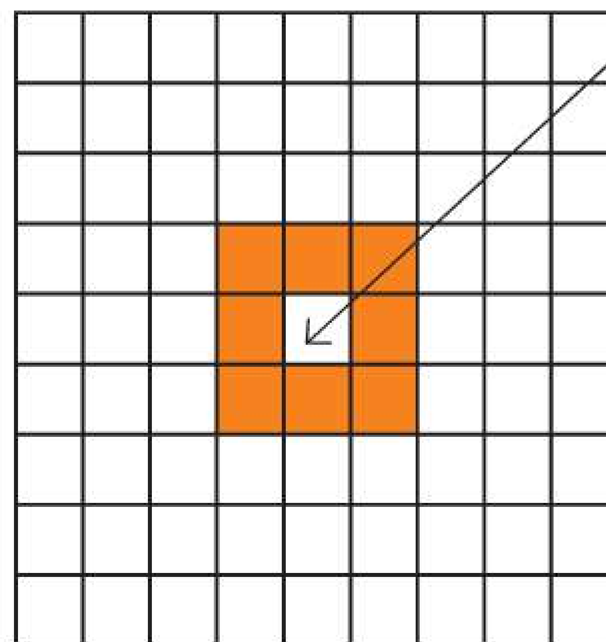
Komórka umiera, gdy ma jednego sąsiada (z samotności) albo gdy ma więcej niż trzech sąsiadów (z przeludnienia), np.:



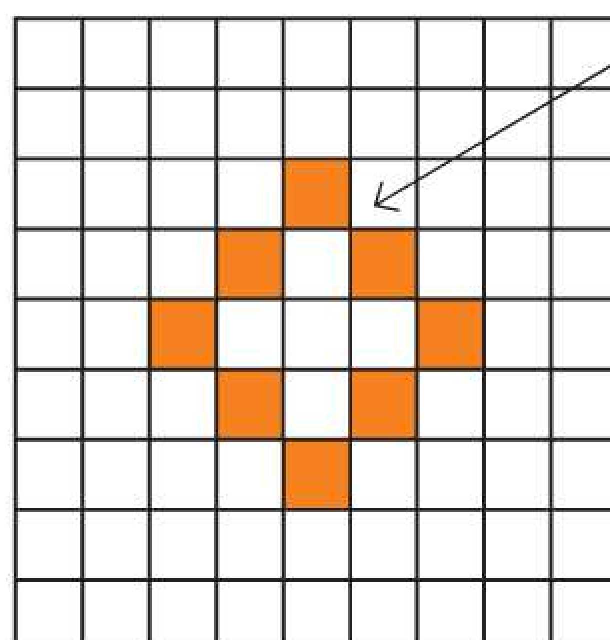
JAK BĘDZIE WYGLĄDAĆ CZWARTE POKOLENIE?



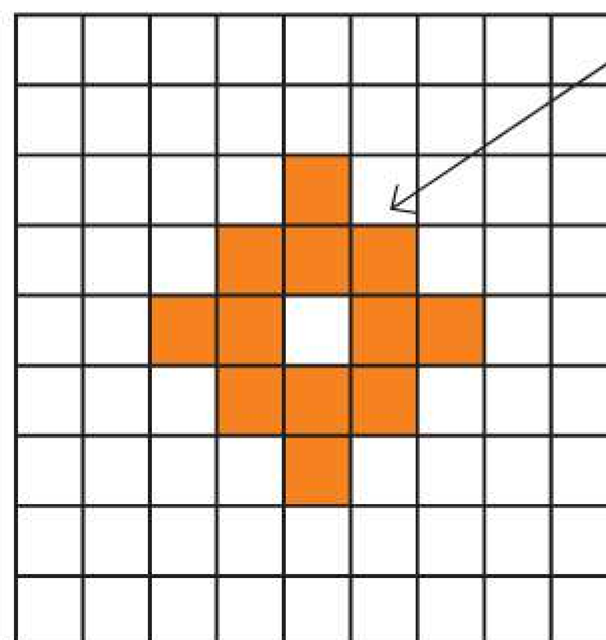
1. Oto planeta i zamieszkujące ją pierwsze pokolenie.



2. Środkowa pomarańczowa komórka umrze z przeludnienia (bo ma czterech sąsiadów), a cztery pozostałe pozostaną przy życiu (bo mają po trzech sąsiadów). Cztery puste komórki ożyją, ponieważ są otoczone przez trzy komórki pomarańczowe.



3. Cztery komórki na środkach boków kwadratu umrą z przeludnienia (mają po czterech sąsiadów). Ożyją cztery komórki, które mają po trzech sąsiadów.



4. Komórki, które mają po dwóch sąsiadów, pozostaną przy życiu. Dodatkowo ożyją cztery komórki, które obumarły w poprzednim pokoleniu, ponieważ mają po trzech sąsiadów.

4.1 Od królików do złotej proporcji

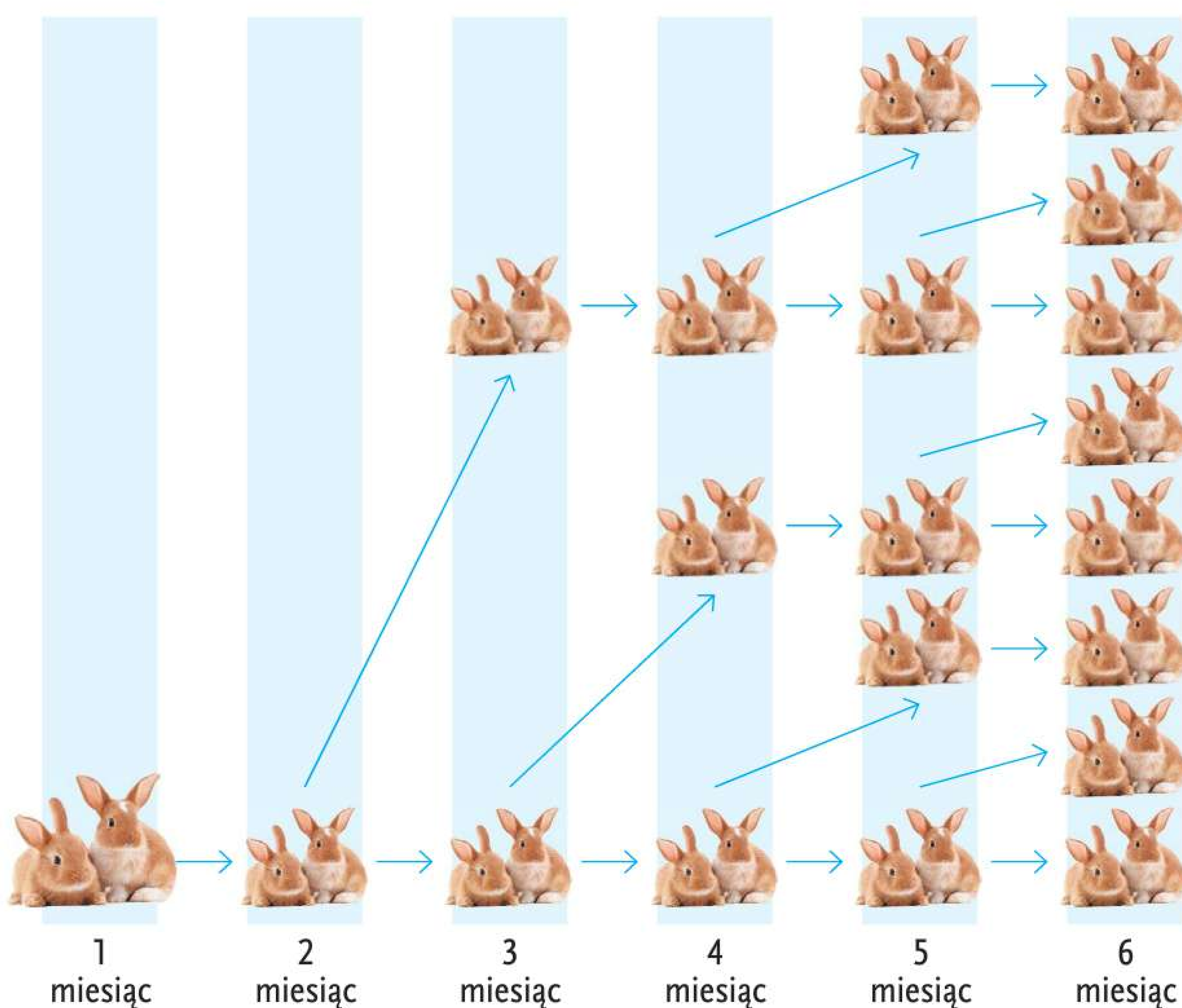
DOWIESZ SIĘ,

- czym są liczby Fibonacciego (czytaj: fibonacziego) i w jaki sposób je wyznaczyć,
- co to jest złota proporcja.

Na początku w klatce jest młoda para królików. Po upływie miesiąca stają się dorosłe i mogą się rozmnażać. Co miesiąc dorosłej parze rodzi się nowa para królików. Ile par królików będzie w tej klatce po roku? Oto problem, który rozważał włoski matematyk Fibonacci w swojej księdze *Liber abaci* (czytaj: liber abaczi; *Księga rachunków*) z 1202 r.

KRÓLIKI I CIĄG FIBONACCIEGO

Przedstaw opisaną sytuację w kolejnych miesiącach. Załóż, że króliki rozmnażają się bez przeszkód i nie umierają.



Rys. 1. Ilustracja zagadnienia populacyjnego opisanego przez Fibonacciego

1. W pierwszym miesiącu jest jedna para młodych królików.
2. Na początku drugiego miesiąca samica może już mieć młode, ale wciąż jest tylko jedna para królików.
3. Na początku trzeciego miesiąca samica rodzi parę królików i teraz już są dwie pary.
4. Na początku czwartego miesiąca samica rodzi drugą parę – w klatce są więc trzy pary królików.
5. Na początku piątego miesiąca pierwsza samica znów rodzi parę, a samica, która urodziła się dwa miesiące temu, rodzi nową parę. Wtedy w klatce jest już pięć par królików (trzy pary z poprzedniego miesiąca i dwie nowe pary).

...

12. W 12. miesiącu w klatce są 144 pary królików.

Powstały w ten sposób ciąg liczb naturalnych, w którym każdy kolejny wyraz (z wyjątkiem dwóch pierwszych) jest sumą dwóch poprzednich wyrazów, nazywany jest ciągiem Fibonacciego. Zwróć uwagę, że liczba dorosłych par i łączna liczba par są opisywane tym samym ciągiem. W obu przypadkach reguła jest więc ta sama: liczbę par w następnym miesiącu można uzyskać przez zsumowanie liczby par w poprzednich dwóch miesiącach. Jeśli jako $F(n)$ oznaczymy n-ty wyraz ciągu, to $F(n) = F(n - 1) + F(n - 2)$, przy czym $F(1) = 1$ oraz $F(2) = 1$.

CIĄG FIBONACCIEGO W ARKUSZU

Arkusz kalkulacyjny pozwala w łatwy sposób uzyskać dużą liczbę wyrazów ciągu Fibonacciego. Można zrobić to wspólnie, korzystając z Dokumentów Google.

Podzielcie się na kilkuosobowe grupy i wybierzcie lidera, który utworzy nowy arkusz, nazwie go Fibonacci i udostępni plik współpracownikom przez kliknięcie przycisku **Udostępnij** i wpisanie adresów mailowych lub przesłanie linku do arkusza. Współpracownicy powinni mieć uprawnienia edytora.

Każdy ze współpracowników będzie mieć na starcie inną liczbę królików (czyli inne pierwsze dwa wyrazy ciągu), dlatego w udostępnionym skoroszycie musi utworzyć własny arkusz i go podpisać. Witek może mieć np. jedną parę królików, Ika – dwie, Wanda – trzy, a Mirek cztery.



Rys. 2. Tworzenie arkuszy

► Zatytułuj pierwszą kolumnę **Ciąg F**, następnie wpisz dwa pierwsze wyrazy ciągu, a w kolejnym wierszu – wzór na zsumowanie dwóch komórek znajdujących się powyżej.

Fibonacci

Plik Edytuj Widok Wstaw

=A2+A3

	A
1	Ciąg F
2	1
3	1
4	=A2+A3

Rys. 3. Arkusz Witka

▶ Skopiuj wzór do 30 kolejnych komórek.

Widać, że królików szybko przybywa...

▶ Wyznacz iloraz dwóch kolejnych wyrazów ciągu – jako tytuł kolumny **B** podaj formułę **F(n + 1)/F(n)**, wpisz w kolejnym wierszu odpowiedni wzór, a następnie skopiuj go do przedostatniego wyrazu i przeanalizuj wyniki.

fx =A3/A2

	A	B
1	Ciąg F	F(n+1)/F(n)
2		=A3/A2
3	1	

Rys. 4. Uzupełnianie kolumny **B** w arkuszu Witka

Wartości są prawie jednakowe i wynoszą ok. 1,618. Mówi to, że w kolejnym pokoleniu jest o ponad 60% więcej królików niż w poprzednim.

Na koniec dodaj kolumnę z odwrotnością tej wielkości, czyli ilorazem **F(n)/F(n + 1)**.

▶ Wpisz odpowiednią formułę w komórce **C2** i przeanalizuj wyniki.

fx =1/B2

	A	B	C
1	Ciąg F	F(n+1)/F(n)	wrotność czyli F(n)/F(n+1)
2	1	1	=1/B2
3	1	2	

Rys. 5. Uzupełnianie kolumny **C** w arkuszu Witka

ZŁOTA PROPORCJA

Wyniki zaskakują. Okazuje się, że końcowe wartości w kolumnach **B** i **C** u wszystkich współpracowników są jednakowe i wynoszą odpowiednio ok. 1,618 i 0,618.

28	785672	1,618033989	0,618033989
29	1271244	**1,618033989**	**0,618033989**
30	2056916		

+ ≣ Witek ▾ Ika ▾ Wanda ▾ Mirek ▾

Rys. 6. Wyniki w arkuszu Mirka

Wartości obu ilorazów u Mirka, który na początku miał cztery pary królików, są takie same jak u Witka, który zaczynał z jedną parą. W dodatku te dwa ilorazy są bardzo podobne, różnią się tylko jedynką przed przecinkiem. Dla dużego n mamy więc zależność $F(n + 1)/F(n) = F(n)/F(n + 1) + 1$.

Jeśli pierwszy iloraz oznaczyć grecką literą φ (fi), to: $\varphi = 1/\varphi + 1$. Te dziwne własności skłaniają do poszukiwania wartości $\varphi \sim$ **1,618033989**. Znajdziesz ją bez problemu w internecie. Okazuje się, że ta liczba to złota proporcja, znana w starożytności i obliczona na podstawie konstrukcji geometrycznej nazywanej **złotym podziałem**. Powstaje on wtedy, gdy odcinek dzieli się na dwie części a i b tak, aby stosunek długości całego odcinka do dłuższej części był taki sam jak stosunek dłuższej części do krótszej, czyli $(a+b)/a = a/b$.

Rys. 7. Złoty podział

Prostokąt zbudowany z zachowaniem zasad złotego podziału – czyli złoty prostokąt – uznawany był za kanon estetycznego kształtu. Do dziś stosuje się go przy projektowaniu fasad budowli i ram obrazów, a nawet ekranów komputerowych.

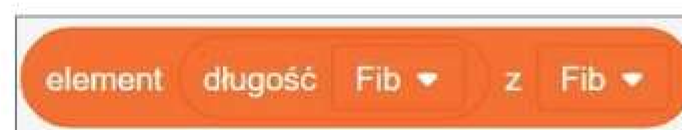
ZADANIA

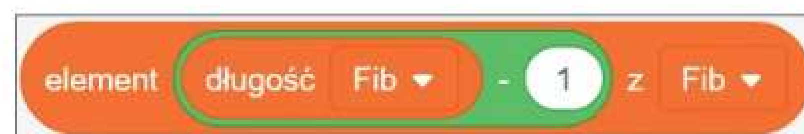
1. Wyszukaj przykłady zastosowania złotej proporcji. Zbierz je w tabeli i uporządkuj według dziedzin.

2. Poszukaj informacji na temat Fibonacciego i jego ciągu. Zbierz grupę zainteresowanych osób i utwórz wspólną prezentację (korzystając z Dokumentów Google lub Office 365). Udostępnij prezentację również nauczycielowi. Znajdź też informacje o tym, gdzie w przyrodzie albo w różnych dziedzinach życia i nauki zaobserwowano podobieństwa do ciągu Fibonacciego. Opisz, na czym one polegają. Oto kilka haseł, żeby łatwiej było ci szukać: przyrost pędów drzewa, przyrost liści na pędach kwiatów, układ łusek w szyszkach i ziaren w słoneczniku, rozmnażanie pszczół, kwadraty Fibonacciego, spirala Fibonacciego.

3. Zaprogramuj obliczanie kolejnych wyrazów ciągu Fibonacciego w wybranym środowisku programistycznym. Ciąg liczb można zapisać w postaci listy (tablicy).











4.2 Kości zostały rzucone

DOWIESZ SIĘ, JAK

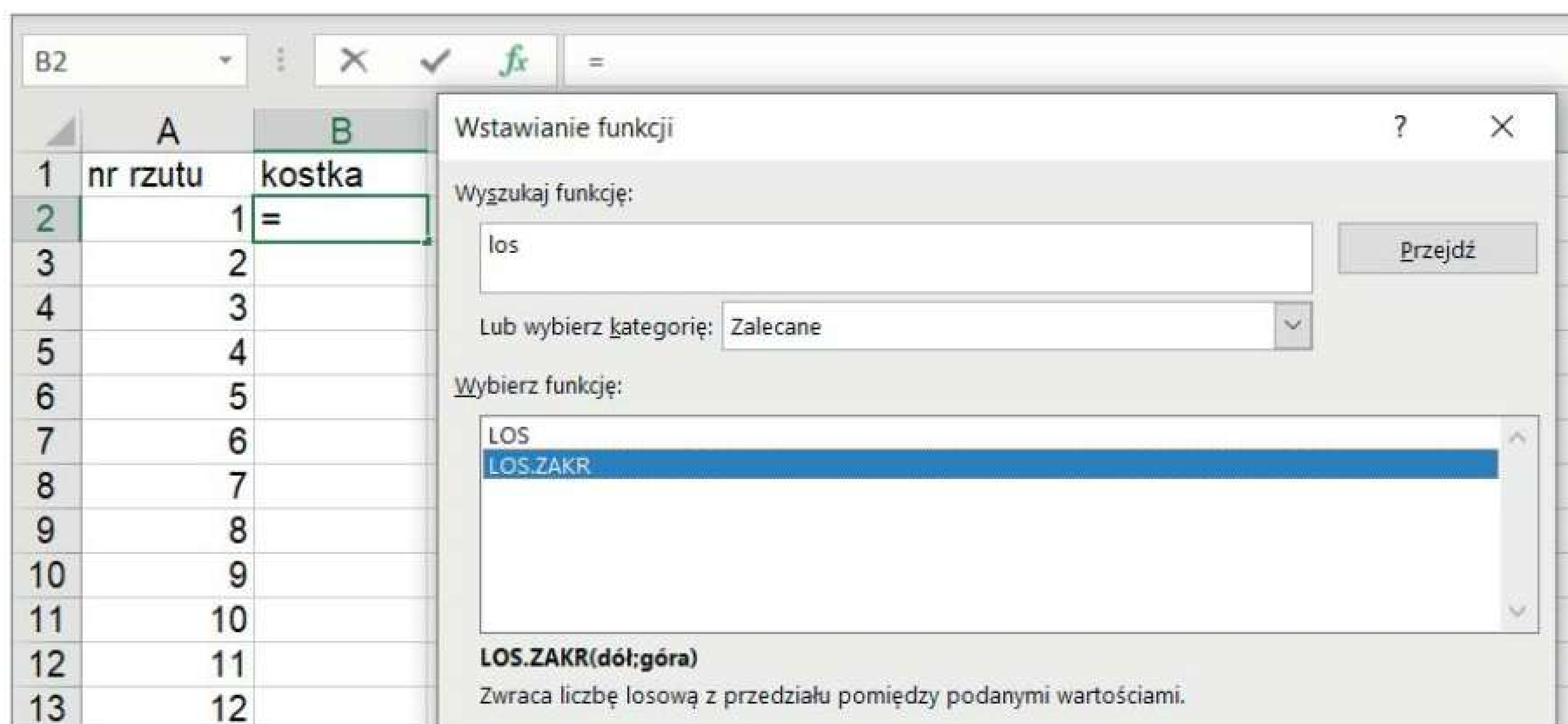
- korzystać z funkcji losowych w arkuszu kalkulacyjnym,
- przeprowadzić symulację procesu o losowym przebiegu.

Wyobraź sobie grę, w której jeden z graczy rzuca dwiema sześciennymi kostkami, a drugi próbuje odgadnąć wynik rzutu. Jaką sumę oczek powinien wybrać? Czy powinien postawić na dwunastkę (dwie szóstki) czy raczej na dziewiątkę (czwórka i piątka lub trójka i szóstka)? Czy wszystkie sumy oczek będą się pojawiały w tej grze równie często?

LOSOWANIE OCZEK

Zauważ, że gdyby gracz rzucał jedną kostką, to możliwych wyników byłoby sześć i każda liczba oczek pojawiałaby się równie często. Przeprowadź symulację 120 takich rzutów kostką w arkuszu kalkulacyjnym. Która funkcja może imitować rzut kostką? Jest to funkcja **LOS.ZAKR**, która podaje losową liczbę całkowitą z przedziału między wskazanymi wartościami.

► Przygotuj w arkuszu kolumny **A** i **B** (na numer zrzutu i wynik), a następnie otwórz okno wstawiania funkcji i wyszukaj funkcje, które zawierają słowo **los**. Wybierz funkcję **LOS.ZAKR** i uzupełnij granice losowania – **1** i **6**.



Rys. 1. Wyszukiwanie funkcji losowej

- Skopiuj formułę do kolejnych 120 komórek i obejrzyj wyniki. Są one nietrwałe, co oznacza, że przy każdym przeliczeniu wartości w komórkach arkusz losuje nowe liczby. Przeliczenie następuje, ilekroć wprowadzisz nową wartość do jakiejkolwiek komórki. Można je również wywołać przez naciśnięcie klawisza funkcyjnego **F9**.

Zrób statystykę wyników – sprawdź, ile razy wylosowane zostały poszczególne wyniki rzutów – a następnie utwórz dla niej wykres kolumnowy.

Tego typu obliczenia wykonuje się za pomocą funkcji **CZĘSTOŚĆ**. Jest ona wstawiana równocześnie do całego zakresu komórek. Aby ją wstawić, trzeba najpierw zaznaczyć obszar, w którym mają się pojawić wyniki, następnie wprowadzić formułę tablicową, a na końcu zatwierdzić tę czynność za pomocą skrótu klawiaturowego **Ctrl+Shift+Enter** (w arkuszu w wersji Microsoft 365 można zatwierdzić operację klawiszem **Enter**). Jeśli wprowadzenie funkcji przebiegnie poprawnie, formuła tablicowa zostanie w arkuszu ujęta w nawiasy klamrowe {} – będą one widoczne w wierszu edycji.

- Kolumnę **F** nazwij **wyniki** i uzupełnij ją wszystkimi możliwymi do uzyskania wynikami.

- W kolumnie **G** oblicz częstość występowania wyników w całym zakresie losowań, czyli w komórkach **B2:B121** – zastosuj funkcję **CZĘSTOŚĆ**.
  - Zaznacz cały obszar przeznaczony na wyniki częstości (komórki **G2:G7**). Wywołaj przyciskiem f_x funkcję **CZĘSTOŚĆ**. Wybierz argumenty funkcji: **Tablica_dane** to wyniki 120 losowań (komórki **B2:B121**), **Tablica_przedziały** to możliwe wyniki rzutu kostką, które zostały wpisane w komórkach **F2:F7**. Zatwierdź funkcję (klawisze **Shift+Ctrl+Enter**) i podsumuj liczbę wyników w kolumnie **G** (powinno ich być 120). Sprawdź, czy formuła tablicowa została ujęta w nawiasy klamrowe.

G2 =CZĘSTOŚĆ(B2:B121;F2:F7)

	A	B	C	D	E	F	G	H	I
1	nr	kostka				wyniki	częstość		
2	1	5				1	F2:F7)		
3	2	3				2			
4	3	5				3			
5	4	3				4			
6	5	5				5			
7	6	2				6			

Argumenty funkcji

CZĘSTOŚĆ

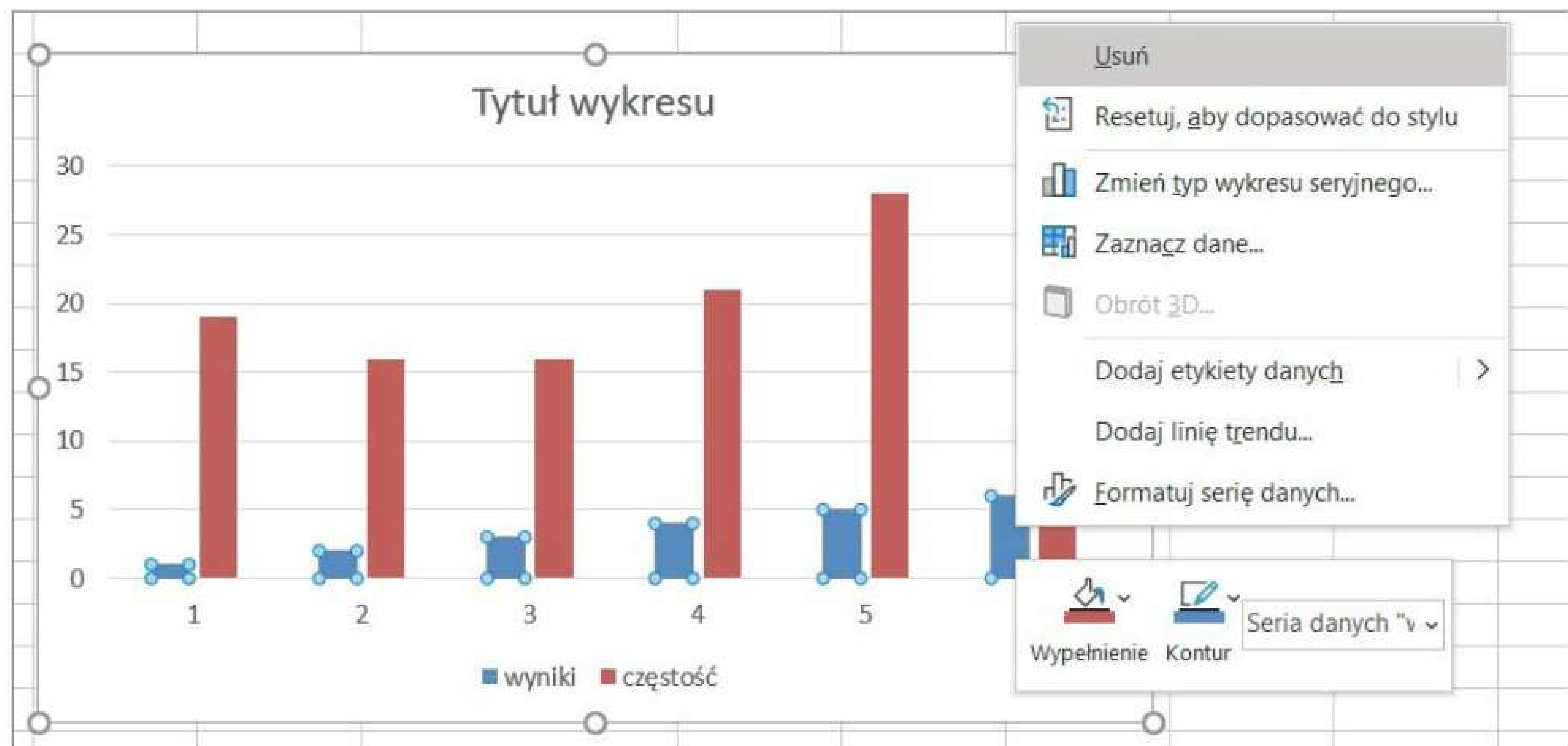
Tablica_dane B2:B121 = {5;3;5;3;5;2;3;6;6;5;3;3;6;4;1;5;2;4;2;3;4;6;2;5;6;2;5;5;

Tablica_przedziały F2:F7 = {1;2;3;4;5;6}

Rys. 2. Wstawianie funkcji **CZĘSTOŚĆ**

- Utwórz wykres częstości – zaznacz tabelę z wynikami i częstością, a potem wybierz **Wstawianie** → **Wstaw wykres kolumnowy**.

Niestety, arkusz nie domyśli się, że chcesz narysować tylko kolumny częstości – musisz więc zaznaczyć i usunąć serię **wyniki**. Rozkład częstości jest w miarę równomierny. Możesz to sprawdzić przez kilkakrotne przeliczanie arkusza (klawisz **F9**).



Rys. 3. Wstawianie wykresu częstości i usuwanie serii **wyniki**

Pseudolosowanie i liczby pseudolosowe

Kto myśli, że komputer faktycznie losuje liczby, jest w błędzie. W rzeczywistości komputer „udaje" realizację procesu losowania z wykorzystaniem rozmaitych metod numerycznych – to dlatego wygenerowane przez niego liczby są nazywane pseudolosowymi. Jedną z metod obliczeniowych wykorzystujących losowość wymyślił Stanisław Ulam. Polski matematyk, który w 1943 r. przyjął również obywatelstwo amerykańskie, opracował metodę Monte Carlo, stosowaną do modelowania matematycznego bardzo złożonych i trudnych obliczeniowo procesów.

RZUT DWIEMA KOSTKAMI

Czas na rzut dwiema kostkami. W arkuszu wystarczy skopiować całą kolumnę **B** z wynikami losowań do kolumny **C**, która będzie reprezentować wyniki losowań dla drugiej kostki. Po odpowiednim oznaczeniu kolumn **B** i **C** trzeba obliczyć w kolumnie **D** sumę oczek na obu kostkach – wpisz do komórki **D2** formułę **=B2+C2** i skopiuj ją do kolejnych komórek kolumny **D**.

	A	B	C	D
1	nr rzutu	kostka 1	kostka 2	
2	1	1	5	=B2+C2
3	2	5	6	

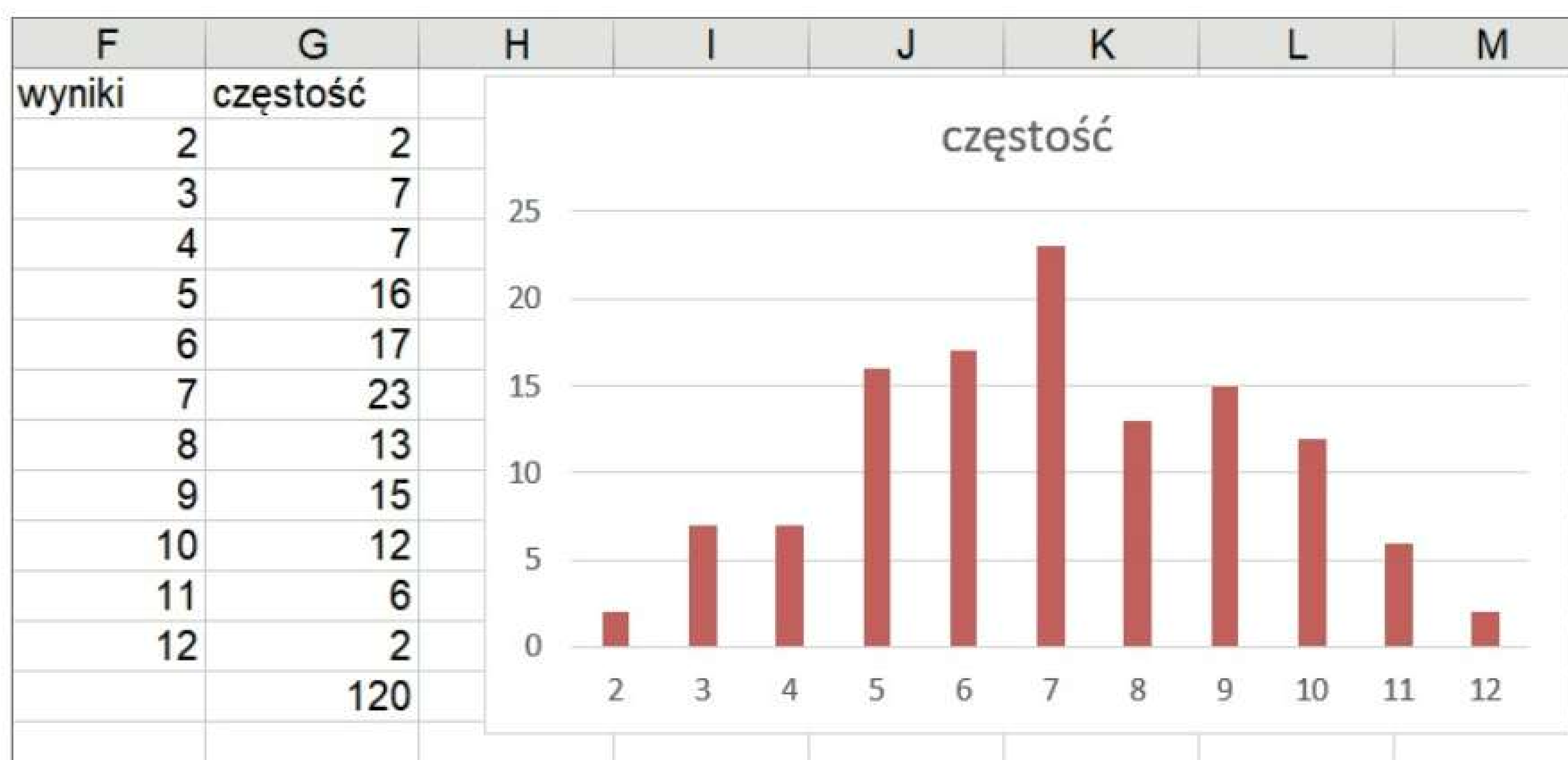
Rys. 4. Obliczanie sumy oczek

Możliwe wyniki rzutu dwiema kostkami to liczby od 2 do 12. Jak poprzednio – wypisz je w kolumnie **F**, a w kolumnie **G** wstaw funkcję **CZĘSTOŚĆ**, żeby obliczyć częstość ich występowania.

	A	B	C	D	E	F	G
1	nr rzutu	kostka 1	kostka 2	suma		wyniki	częstość
2	1	1	6	7		2	=
3	2	4	5	9		3	
4	3	3	3	6		4	
5	4	3	2	5		5	
6	5	3	4	7		6	
7	6	1	2	3		7	
8	7	3	1	4		8	
9	8	4	2	6		9	
10	9	6	1	7		10	
11	10	6	1	7		11	
12	11	2	3	5		12	
13	12	3	4	7			
14	13	5	5	10			
15	14	5	4	9			
16	15	6	3	9			

Rys. 5. Fragment tabeli dla rzutu dwiema kostkami

Przyjrzyj się otrzymanym wynikom. Wydaje się, że częściej występują wyniki środkowe (6, 7, 8), natomiast rzadziej – wyniki skrajne (2, 12). Sprawdź to za pomocą wykresu kolumnowego.



F	G
wyniki	częstość
2	2
3	7
4	7
5	16
6	17
7	23
8	13
9	15
10	12
11	6
12	2
	120

Rys. 6. Wstawianie wykresu częstości

Jak wytłumaczyć nierównomierność rozkładu częstości? Trzeba przeanalizować, jak mogą powstać poszczególne wyniki. Najczęściej występujący wynik 7 można otrzymać wtedy, kiedy na jednej kostce będzie 1, a na drugiej – 6 (lub odwrotnie), na jednej 2, a na drugiej... Spróbuj wypisać wszystkie możliwości. Ile ich jest? Arkusz pomoże ci w tych obliczeniach.

- Nadaj pierwszemu arkuszowi nazwę, np. **rzuty kostką**, i przejdź do nowego arkusza w tym samym skoroszycie. Przedstawisz w nim wszystkie możliwości rzutu dwiema kostkami. Musisz zrobić tabliczkę dodawania od 1 do 6.
- W drugim wierszu wypisz, począwszy od komórki **C2**, wszystkie możliwe wyniki dla pierwszej kostki: 1, 2... Następnie w kolumnie **B**, począwszy od komórki **B3**, wypisz wszystkie możliwe wyniki dla drugiej kostki.
- Zakres komórek **C2:H2**, zawierający wyniki dla pierwszej kostki, nazwij np. **kostka1**, a zakres komórek **B3:B8**, zawierający wyniki dla drugiej kostki – np. **kostka2**. Po co te nazwy? Żeby łatwo było wpisać formułę sumowania oczek **=kostka1+kostka2**. Wpisz ją do komórki **C3** i na koniec skopiuj formułę do pozostałych komórek tabeli.

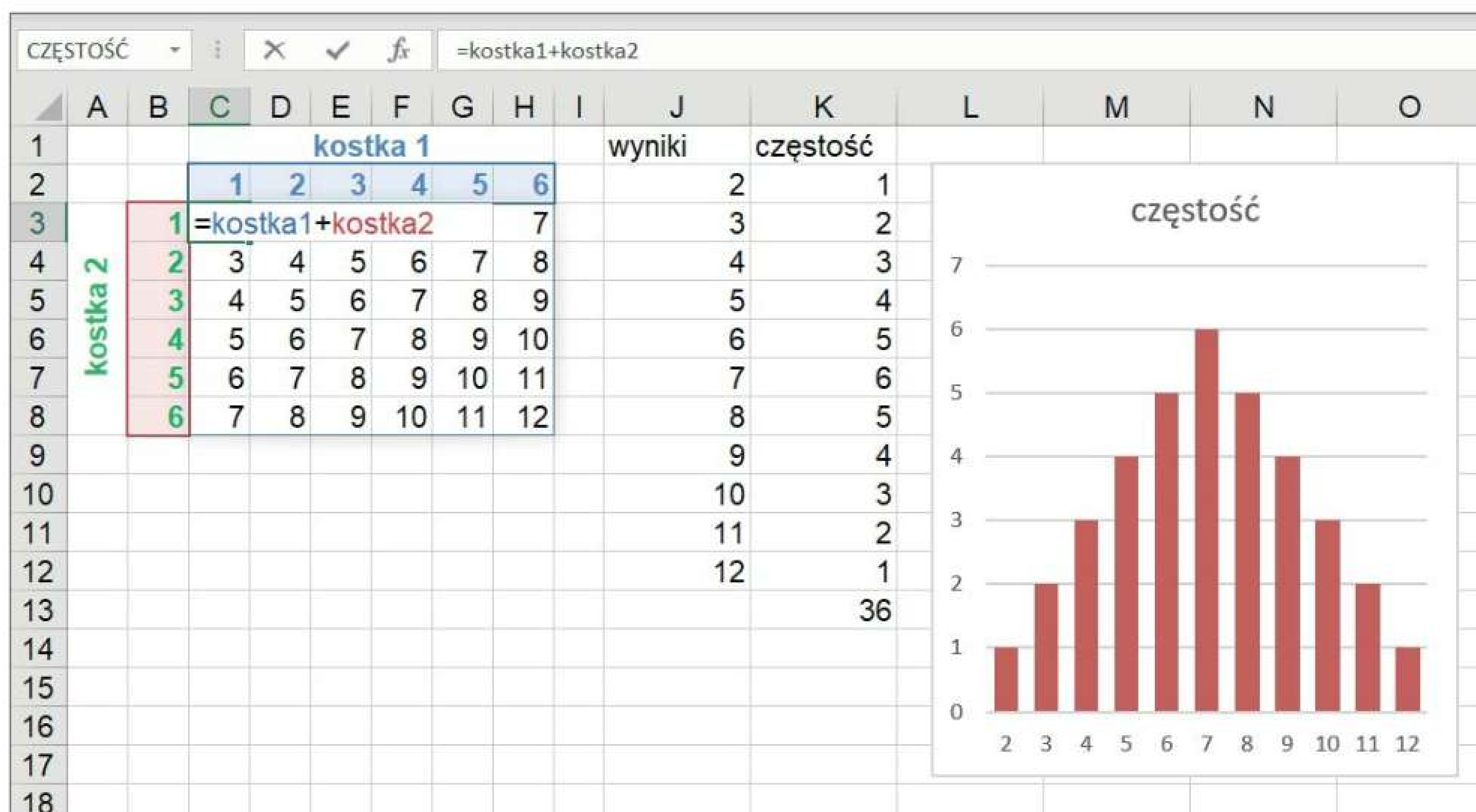
kostka1 | fx 1

	A	B	C	D	E	F	G	H	I
1			kostka 1						
2			1	2	3	4	5	6	
3		1							
4		2							

Rys. 7. Tworzenie tabliczki dodawania

Zrobione? No to masz komplet 36 możliwych wyników rzutów.

- Sprawdź, ile razy występuje wynik sumaryczny 7, a ile razy – wynik 2.
- Oblicz częstość występowania możliwych wyników i przedstaw otrzymany rozkład w postaci wykresu częstości.



	A	B	C	D	E	F	G	H	I	J	K
1			kostka 1							wyniki	częstość
2			1	2	3	4	5	6		2	1
3		1	=kostka1+kostka2					7		3	2
4		2	3	4	5	6	7	8		4	3
5		3	4	5	6	7	8	9		5	4
6		4	5	6	7	8	9	10		6	5
7		5	6	7	8	9	10	11		7	6
8		6	7	8	9	10	11	12		8	5
9										9	4
10										10	3
11										11	2
12										12	1
13											36

Rys. 8. Wyniki rzutów i rozkład ich częstości – teoretyczne uzasadnienie przeprowadzonego doświadczenia

ZADANIA

1. W niektórych grach występują kostki o innej liczbie ścian niż sześć: ośmiościenna, dziesięciościenna, dwunastościenna i dwudziestościenna. Ściany kostek są ponumerowane. Wszystkie kostki mają tę własność, że prawdopodobieństwo wyrzucenia każdej liczby oczek jest jednakowe. Wybierz jedną z takich kostek i zaplanuj grę polegającą na rzucaniu dwiema kostkami, sumowaniu oczek i przewidywaniu najczęściej występującego wyniku (sumy). Podaj wyniki wygrywające, czyli te, które będą występowały najczęściej.



2. Skorzystaj z pomocy arkusza kalkulacyjnego i poznaj funkcję pseudolosową **LOS**. Generuje ona liczbę losową z przedziału **⟨0, 1)**, co oznacza, że wylosowana liczba rzeczywista musi być równa 0 albo większa od 0 i jednocześnie mniejsza od 1. Czy dzięki tej funkcji można uzyskać wartości potrzebne przy obliczeniach prowadzonych na lekcji (liczby całkowite od 1 do 6)? Jak można je uzyskać za pomocą funkcji **LOS**? Może ci się przydać również funkcja **ZAOKR.DO.CAŁK** (zaokrągla liczbę w dół do najbliższej liczby całkowitej).

3. Napisz program w języku Python, który spowoduje wypisanie 24 liczb losowych z zakresu od 1 do 6, czyli zasymuluje 24 rzuty kostką sześcienną. Wykorzystaj funkcję **`randint(zakres)`** z modułu **`random`**. Zakres to oczywiście od 1 do 6. Uruchom program kilkakrotnie, przeglądając za każdym razem wyniki.

4.3 Fraktale w Scratchu i w Pythonie

DOWIESZ SIĘ,

- czym jest fraktal,
- jak narysować drzewo binarne w Scratchu i w Pythonie.

O fraktalach często mówi się, że stanowią uporządkowany chaos. Można dostrzec je wszędzie: w naturze, sztuce, technice, architekturze. Komputerowo konstruuje się je na podstawie algorytmów matematycznych. Zobacz, jak stworzyć jeden z prostszych fraktali w Scratchu i Pythonie.

CZYM SĄ FRAKTALE

Fraktale to wymyślone przez matematyków figury geometryczne, składające się z mniejszych elementów, które wyglądają tak samo jak cała figura. Ich ciekawe kształty i specyficzne własności sprawiły, że fraktale rozpowszechniły się w grafice komputerowej – stosując stosunkowo proste programy oparte na rekurencji, można szybko stworzyć modele drzew, liści czy płatków śniegu oraz całe krajobrazy. Ale to nie wszystko – geometrię fraktalną wykorzystuje się również w radioelektronice przy budowie anten (np. w telefonach komórkowych), do opisu i przewidywania zjawisk przyrodniczych (np. trzęsień ziemi) oraz reakcji chemicznych, a nawet do przewidywania zmian kursów giełdowych.





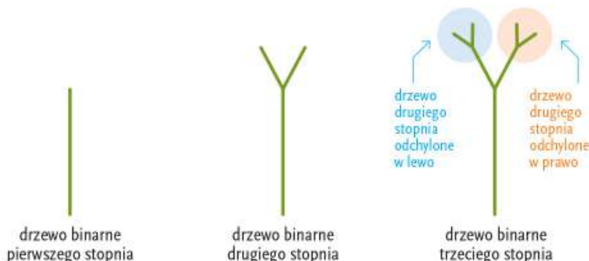
Rys. 1. Przykłady fraktali

Teraz narysujesz swój pierwszy fraktal – drzewo binarne. Nazwano je tak, ponieważ z każdego jego rozgałęzienia wyrastają dwie podobne gałęzie.

F

DRZEWO BINARNE

Drzewo binarne charakteryzują dwie liczby: **stopień** (liczba poziomów) oraz **wielkość** (długość pnia). Poniższe drzewo składa się z trzech poziomów. Pierwszy to pień, drugi – dwie wyrastające z niego gałęzie, a trzeci – podwójne rozgałęzienia tych gałęzi. Gałęzie zawsze wyrastają z pnia pod kątem 30° i mają długość równą połowie długości pnia (zauważ, że gałąź, z której wyrastają kolejne gałęzie, stanowi jednocześnie pień drzewa niższego stopnia).



Rys. 1. Etapy powstawania drzewa binarnego trzeciego stopnia

Zaplanuj konstrukcję drzewa binarnego. Aby ułatwić sobie zadanie, przyjmij, że pień o pewnej długości to element podstawowy drzewa. Wyrastają z niego dwa identyczne drzewa o pniu dwa razy krótszym od pnia drzewa pierwotnego. Jedno z drzew jest odchylone od pnia o 30° w lewo, a drugie o 30° w prawo (czyli drugie drzewo jest odchylone o 60° w prawo od pierwszego).

Plan rysowania drzewa binarnego może zatem wyglądać następująco:

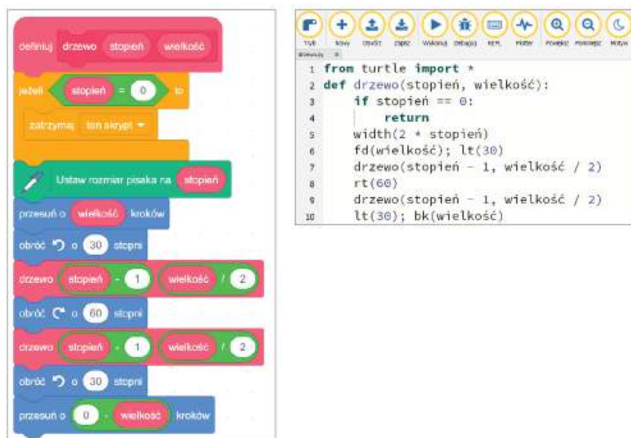
1. Ustal warunki – wprowadź stopień i wielkość drzewa (długość pnia).
   - Jeśli stopień równa się 0, zatrzymaj wykonywanie skryptu.
   - Jeśli stopień jest różny od 0, kontynuuj wykonywanie skryptu.
2. Narysuj drzewo pierwszego stopnia – przejdź naprzód o długość pnia.
3. Skręć w lewo o 30°.
4. Narysuj kolejne drzewo – wywołanie drzewa z parametrami `stopień - 1` i `wielkość/2`.
5. Skręć w prawo o 60°.
6. Narysuj kolejne drzewo – wywołanie drzewa z parametrami `stopień - 1` i `wielkość/2`.
7. Skręć w lewo o 30° i przejdź wstecz o długość pnia, aby znaleźć się u podstawy pnia.

Po co wracać do podstawy pnia? Wiąże się to z wywołaniami rekurencyjnymi. Podczas rysowania drzew w krokach 4 i 6 obowiązywało założenie, że rysowanie kończy się w tym samym miejscu, w którym się zaczynało.

Pozostało przenieść algorytm do Scratcha i Pythona. Aby drzewo wyglądało bardziej realistycznie, możesz dodatkowo uzależnić grubość pisaka od stopnia drzewa – odcinki na kolejnych poziomach będą coraz cieńsze.

▶ Zbuduj skrypt w wybranym środowisku.

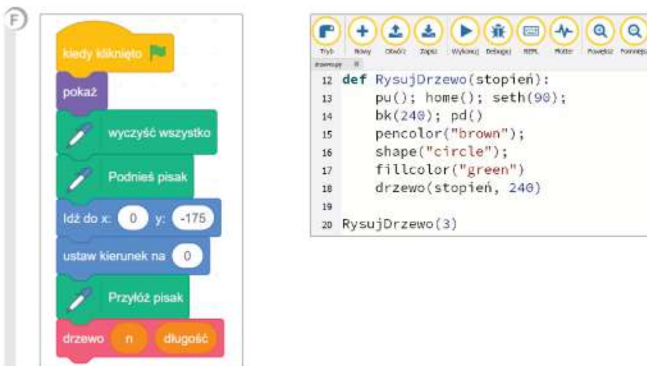
- W Scratchu zdefiniuj nowy blok **drzewo** z danymi wejściowymi **stopień** i **wielkość**.
- W Pythonie zaimportuj moduł **turtle** i zdefiniuj funkcję **drzewo** z parametrami **stopień** i **wielkość**. Stopień to liczba poziomów, a wielkość to długość pnia. Za grubość pisaka odpowiada polecenie **width()**.



Rys. 2. Skrypt drzewa binarnego w Scratchu i w Pythonie

▶ Wywołaj odpowiednio blok lub funkcję odpowiedzialne za rysowanie drzewa.

- W Scratchu zbuduj startowy skrypt zielonej flagi. Utwórz zmienne **n** i **długość** oraz za pomocą suwaków nadaj im wartości, np. 3 i 240.
- W Pythonie zapisz definicję funkcji z parametrem, w której ustawisz żółwia tak, aby zapewnić mu miejsce do narysowania drzewa, i zmienisz kolor pisaka. Polecenie **home()** powoduje powrót żółwia (trójkącika) na środek okna, polecenie **seth(90)** ustala kierunek rysowania (do góry), a polecenie **shape("circle")** zmienia kształt żółwia na kółko.

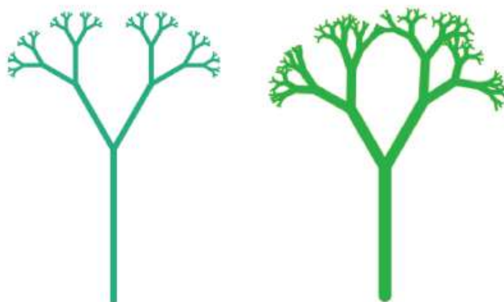


Rys. 3. Skrypt wywołujący drzewo binarne w Scratchu i Pythonie

Po przetestowaniu skryptów (i w razie potrzeby naniesieniu poprawek) zmień w bloku (funkcji) `drzewo` kąt rozchylenia gałęzi na 90°. Czy umiesz przewidzieć, co się stanie?

DRZEWO BINARNE LOSOWE

Skonstruowane drzewo nie przypomina prawdziwego drzewa, ponieważ jest zbyt regularne. Jak to poprawić? Trzeba zaprogramować rysowanie drzewa binarnego, którego gałęzie mają różne długości i rozchylają się pod różnymi kątami.



Rys. 4. Drzewo binarne regularne (z lewej) i drzewo binarne losowe (z prawej)

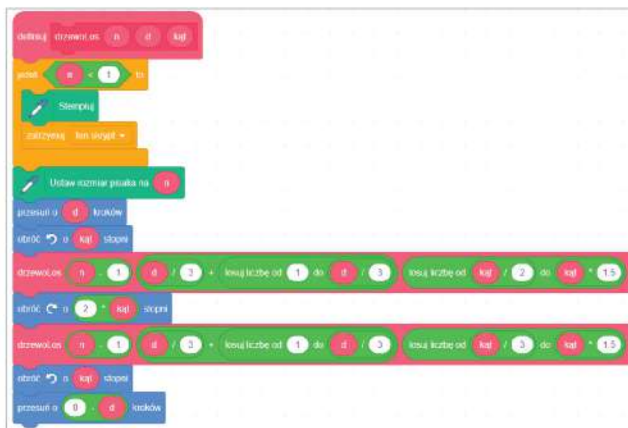
▶ Otwórz zrealizowany w Scratchu projekt **drzewo binarne losowe** (**scratch.mit.edu/projects/15453604**) i sprawdź rezultat wywołania **drzewoLos** z parametrami: **8** (stopień), **120** (wielkość) i **35** (kąt w stopniach). Jaki rysunek powstanie po kolejnym uruchomieniu **drzewoLos** z parametrami **8**, **120**, **35** (takimi samymi przy każdym kolejnym wywołaniu)?

Czym różni się blok **drzewoLos** od bloku **drzewo**?

Po pierwsze, przybył trzeci parametr, a mianowicie kąt rozchylenia gałęzi (o nazwie **kąt**); dzięki wprowadzeniu tego parametru można rysować drzewa bardziej lub mniej rozłożyste.

Po drugie, zmieniły się wywołania poddrzew – mają one teraz nieco inne parametry; wielkość poddrzewa i kąt rozchylenia gałęzi przyjmują teraz wartości losowe.

Czy bardzo skomplikowało to skrypt? Nie. Większość zmian polega na nowych, bardziej złożonych obliczeniach wartości parametrów z użyciem bloku **losuj** w Scratchu i funkcji losowej **randint** w Pythonie, którą trzeba zaimportować z modułu **random**.



Rys. 5. Skrypt drzewa binarnego losowego w Scratchu

▶ Przenieś skrypt drzewa binarnego losowego do Pythona. Zastosuj funkcję losową **randint**, którą trzeba zaimportować z modułu **random**.

```
from turtle import *
from random import randint

def drzewoLos(n, a, k):
    if n==0:
        stamp()
        return
    width(2 * n)
    fd(a); lt(k)
    drzewoLos(n-1, a/2+randint(int(-a/3), int(a/6)), 2*k/randint(1,3))
    rt(2*k)
    drzewoLos(n-1, a/2+randint(int(-a/3), int(a/6)), 2*k/randint(1,3))
    lt(k); bk(a)

def RysujDrzewoLos(stopień):
    pu(); home(); seth(90); bk(240); pd()
    pencolor("brown"); shape("circle"); fillcolor("green")
    drzewoLos(stopień, 240, 30)

RysujDrzewoLos(6)
```

Rys. 6. Skrypt drzewa binarnego losowego w Pythonie

A teraz spróbuj zmodyfikować odpowiednio blok i funkcję **`drzewoLos`**, tak aby twoje drzewa binarne losowe jak najbardziej przypominały prawdziwe drzewa.

ZADANIA

1. Zmodyfikuj funkcję **`drzewo`** tak, aby kąt rozchylenia gałęzi był trzecim parametrem. Narysuj kilka drzew o różnych kątach.
2. Ile gałęzi (odcinków) znajduje się na kolejnych poziomach drzewa? Jak ta liczba zależy od stopnia drzewa? Z ilu gałęzi (odcinków) składa się całe drzewo? Jak liczba wszystkich gałęzi (odcinków) drzewa zależy od jego stopnia?
3. Drzewo Pitagorasa to fraktal podobny do drzewa binarnego. Znajdź w internecie rysunki przedstawiające drzewa Pitagorasa i zestaw je w tabeli z rysunkami drzew binarnych uzyskanymi podczas tej lekcji. Spróbuj zaprogramować drzewo Pitagorasa w Scratchu lub Pythonie.

4.4 Fraktale w smartfonie

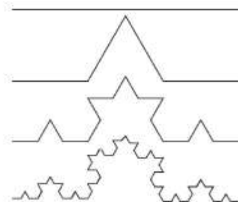
DOWIESZ SIĘ, JAK

- utworzyć aplikację rysującą fraktale w smartfonie za pomocą środowiska App Lab,
- rysować takie fraktale, jak płatek Kocha i trójkąt Sierpińskiego.

Podczas tej lekcji skonstruujesz dwa kolejne fraktale – płatek Kocha i trójkąt Sierpińskiego. Tym razem posłużysz się nowym środowiskiem, wykorzystującym język JavaScript, znany ci już z lekcji poświęconych tworzeniu interaktywnych stron WWW.

PŁATEK KOCHA

Jednym z najstarszych fraktali jest krzywa Kocha, którą opisał szwedzki matematyk Helge von Koch (1870–1924).



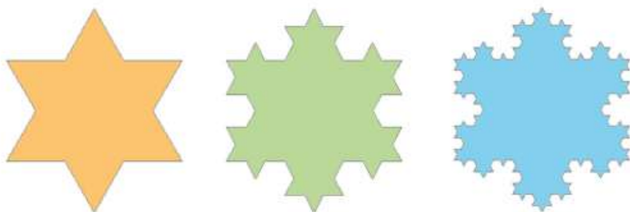
Rys. 1. Transformacja krzywej Kocha od stopnia 0. do stopnia 3.

Trzy krzywe, odpowiednio połączone, tworzą tzw. płatek Kocha, który przypomina płatek śniegu.



Rys. 2. Mikroskopowe powiększenie płatków śniegu

F ▶ Przyjrzyj się uważnie płatkom z rys. 3. Na czym polega ich fraktalność?



Rys. 3. Płatki Kocha stopnia 1., 2. i 3.

Jak skonstruować płatek Kocha? Należy narysować trójkąt równoboczny. Każdy z jego boków, który ma być krzywą Kocha, trzeba następnie podzielić na trzy równe części, a potem w miejsce środkowej części wstawić wystający „ząbek", utworzony przez dwa odcinki o długości trzy razy mniejszej od długości pierwotnego boku trójkąta, które wraz z usuniętą środkową częścią tworzyłyby trójkąt równoboczny. Operację tę należy powtórzyć z każdym z coraz mniejszych boków.

▶ Otwórz zrealizowany w Scratchu projekt **platek** (**scratch.mit.edu/projects/10061548**) i sprawdź, jak zmienia się wygląd płatka wraz ze zmianą jego stopnia oraz długości boku. Przeanalizuj też skrypty wykreślające bok płatka Kocha oraz sam płatek – skrypt rysowania boku jest oczywiście rekurencyjny, bok dzieli się, aż odcinki będą tak krótkie, że nie da się ich narysować.

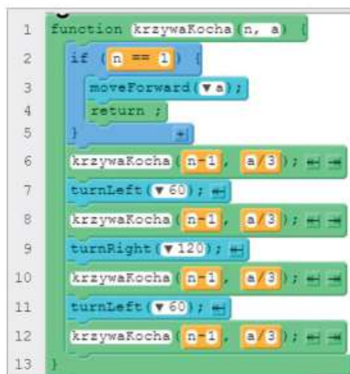
Czas stworzyć aplikację do wykreślania płatków Kocha na smartfonie lub tablecie. Tym razem będziesz pracować w środowisku programowania wizualnego **App Lab**, dostępnym w serwisie organizatora akcji Godzina Kodowania – **code.org**.

▶ Otwórz stronę **https://code.org/educate/applab** i zaloguj się, by rozpocząć nowy projekt (zarejestruj się lub użyj konta Google albo konta wykorzystywanego podczas Godziny Kodowania).



Rys. 4. Fragment głównej strony środowiska App Lab po zalogowaniu się

- Jeśli potrzebujesz objaśnień na temat środowiska, skorzystaj z kursu **Wprowadzenie do App Lab**.
- Zaprogramuj płatek Kocha w środowisku App Lab.
- Przetestuj program – w ostatnim wywołaniu rysującym płatek Kocha zmieniaj stopień płatka w zakresie od 1 do 7. Jeśli stwierdzisz, że program nie działa poprawnie, przeanalizuj poniższy zrzut oraz zapis reszty kodu w postaci tekstowej, wprowadź poprawki i ponownie przeprowadź testy.



Rys. 5. Skrypt rysujący krzywą Kocha – widok blokowy

Zauważ, że zapis blokowy możesz jednym kliknięciem zamienić na zapis tekstowy (i odwrotnie) za pomocą przycisku **Pokaż tekst** (**Pokaż bloki**). Reszta kodu w wersji tekstowej wygląda następująco:

```
function platekKocha(n, a) {
  for (var i = 0; i < 3; i++) {
    krzywaKocha(n, a);
    turnRight(120);
  }
}
show();
penUp();
moveTo(10, 350);
turnTo(30);
penDown();
platekKocha(1, 300);
```

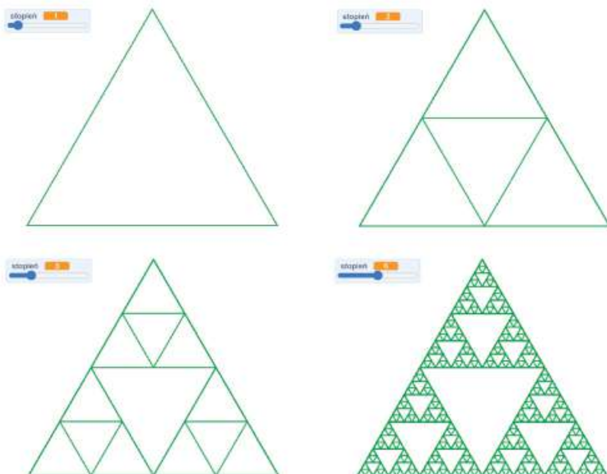
Rys. 6. Skrypt rysujący płatek Kocha – widok tekstowy

Aplikację utworzoną w App Lab możesz otworzyć w smartfonie. Wystarczy kliknąć przycisk **Udostępnij** i zeskanować kod QR lub skopiować podany odnośnik i przesłać go sobie np. e-mailem.

TRÓJKĄT SIERPIŃSKIEGO

Kolejny fraktal opisał polski matematyk Wacław Sierpiński (1882–1969). Trójkąt zwany też uszczelką Sierpińskiego powstał jako konstrukcja pomocnicza do wyprowadzenia twierdzenia o krzywej, która rozgałęzia się w każdym miejscu.

Przeanalizuj projekt **trójkąt Sierpińskiego** zrealizowany w Scratchu (**scratch.mit.edu/projects/15462012**). Jeśli ustawisz suwakiem stopień 1., zostanie narysowany jeden duży trójkąt. Jeśli ustawisz stopień 2., duży trójkąt zostanie podzielony na cztery mniejsze trójkąty o bokach równych połowie boku dużego trójkąta. Zauważ, że każdy z narożnych trójkątów jest też trójkątem Sierpińskiego. Można powiedzieć, że trójkąt Sierpińskiego składa się z trzech trójkątów Sierpińskiego niższego stopnia o bokach dwa razy mniejszych od boku dużego trójkąta i umieszczonych w jego narożnikach.



Rys. 7. Trójkąty Sierpińskiego stopnia 1., 2., 3. i 6.

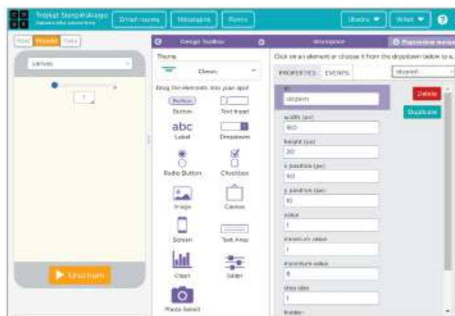
Zastanów się, jak zaprogramować trójkąt Sierpińskiego w JavaScripcie. Zwróć uwagę, że funkcja rysująca trójkąt powinna zawierać dwa parametry: `stopień` i `bok`. Powinna też trzy razy wywołać rekurencyjnie samą siebie z parametrami: `(stopień - 1)` oraz `(bok/2)`. Ponieważ w języku JavaScript nie ma prostej pętli `powtórz`, trzeba w tym celu wykorzystać pętlę `for` (podobnie jak w Pythonie).

- Utwórz w środowisku App Lab projekt rysujący trójkąt Sierpińskiego.
- Przetestuj program. Jeśli stwierdzisz, że nie działa poprawnie, przeanalizuj poniższy zrzut.

```
function trojkatS(n, a) {
  if (n == 0) {
    return;
  }
  for (var i = 0; i < 3; i++) {
    trojkatS(n - 1, a / 2);
    moveForward(a);
    turnRight(120);
  }
}
penUp();
moveTo(10, 400);
turnTo(30);
penDown();
trojkatS(1, 300);
```

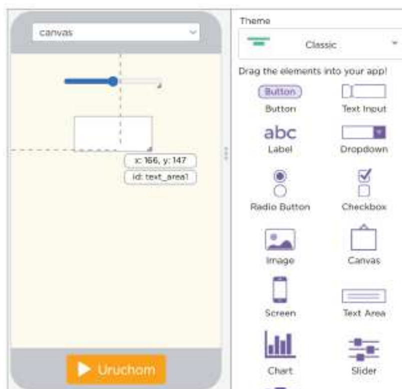
Rys. 8. Skrypt rysujący trójkąt Sierpińskiego

W projekcie App Lab można nie tylko układać program z bloków, lecz także tworzyć pola tekstowe (**TextArea**) i suwaki (**Slider**). Można też zmieniać własności obiektów, np. kolor ekranu (**Canvas**). Aby utworzyć obiekt, trzeba kliknąć przycisk **Projekt** nad oknem przedstawiającym ekran smartfona.



Rys. 9. Tryb projektowy App Lab

F ▶ Przejdź do trybu projektowego App Lab i przeciągnij na ekran smartfona suwak oraz pole tekstowe, a następnie zmodyfikuj ich właściwości.



Rys. 10. Tworzenie obiektu w środowisku App Lab

Po utworzeniu obiektu można do skryptu wstawić zdarzenie definiujące, np. co się stanie, gdy klikniemy obiekt (karta **EVENTS**).

Oprogramowanie pola tekstowego **`ile`** oraz suwaka **`stopień`** może wyglądać następująco:

```
setText("ile", "1");//ustawienie wartości pola tekstowego na 1
onEvent("stopień", "change", function(event) {
  penColor("#fdfaeb");//te trzy linie wymazują poprzedni rysunek
  moveTo(160, 280);//#fdfaeb to kolor tła ekranu
  dot(240);
  moveTo(10, 400);
  turnTo(30);//ustawienia początkowe żółwia
  penColor("black");
  setText("ile", getNumber("stopień"));
  trojkatS(getNumber("stopień"), 300);//rysowanie trójkąta
});//Sierpińskiego stopnia ustawionego na suwaku "stopień"
```

Rys. 11. Tworzenie pola tekstowego i suwaka w środowisku App

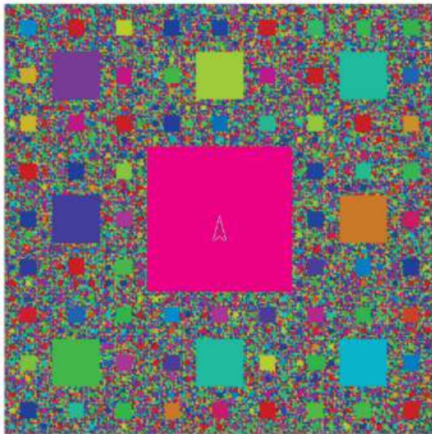
Jeżeli interesuje cię programowanie, zbadaj możliwości środowiska App Lab i utwórz własny projekt. Bloki powinny ci ułatwić pracę z JavaScriptem. Poza tym na głównej stronie App Lab znajdziesz mnóstwo przydatnych materiałów, m.in. kurs wprowadzający do posługiwania się środowiskiem (**Wprowadzenie do App Lab**), przykładowe projekty na start (**Starter Projects**), krótkie filmy o tworzeniu aplikacji App Lab (**Demos for creating apps**), a także bibliotekę wideo (**Video library**).

ZADANIA

1. Płatki mają pewną ciekawą własność: im większy jest ich stopień, tym szybciej rośnie obwód figury i tym wolniej rośnie jej pole, które w końcu, podczas rysowania z bardzo dużą dokładnością, w zasadzie osiąga skończoną wartość (podobnie jest z linią brzegową wyspy – im bliższa odległość, z której ją oglądamy, tym więcej dostrzegamy szczegółów, ale powierzchni wyspy nie przybywa znacząco). Oblicz obwód płatka Kocha dla stopni od 1. do 5. Przyjmij, że pierwotny bok ma długość 1. Wyniki przedstaw w tabeli i krótko podsumuj.

2. Zmodyfikuj funkcję `krzywaKocha` tak, aby narysować płatek wklęsły, w którym „ząbki" na bokach trójkąta są odwrócone do jego wnętrza.

3. Wacław Sierpiński oprócz trójkąta stworzył jeszcze kwadratowy fraktal nazywany dywanem Sierpińskiego. Dywan powstaje w wyniku podzielenia kwadratu na dziewięć równych części, usunięcia środkowego kwadratu i ponownego zastosowania tej procedury do każdego z pozostałych ośmiu kwadratów. Po kolejnych takich operacjach dywan będzie miał tyle „dziur", którymi są usunięte kwadraty, że jego powierzchnia będzie bliska zeru. Znajdź w internecie rysunki dywanów Sierpińskiego i zestaw je w tabeli obok trójkątów tego samego stopnia.



Kolorowy dywan Sierpińskiego stopnia 6.

4. Wykorzystując umiejętności zdobyte na poprzedniej lekcji, zaprogramuj rysowanie jednego z fraktali omawianych podczas tych zajęć w Scratchu lub Pythonie.

4.5 Laboratorium pomiarów

DOWIESZ SIĘ, JAK

- wygląda płytka micro:bit (czytaj: majkrobit) i jak ją zaprogramować,
- wykorzystać smartfon do wykonywania pomiarów,
- mierzyć światło odbite od kolorowych powierzchni.

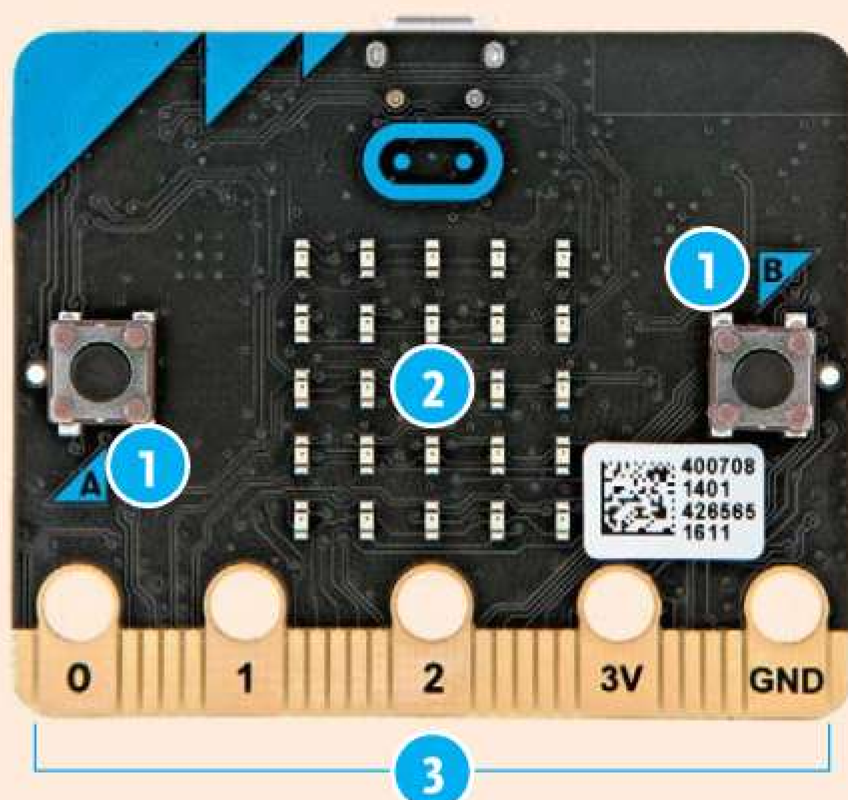
Komputery i smartfony umożliwiają wykonywanie pomiarów. Poznasz teraz dwa sposoby wykonania pomiaru światła odbitego: za pomocą płytki micro:bit i z wykorzystaniem aplikacji Phyphox (czytaj: fyfoks).

POMIARY Z PŁYTKĄ MICRO:BIT

Czym jest micro:bit? To mały programowalny mikrokomputer, który może być wykorzystany do zbudowania wielu urządzeń – od robotów do instrumentów muzycznych. Płytka ma szerokość ok. 5 cm i została przygotowana przez BBC, żeby zachęcić uczniów do eksperymentowania z elektroniką, tworzenia własnych urządzeń i ich programowania. Możliwości jest bardzo wiele. Podczas lekcji przeprowadzisz za pomocą płytki micro:bit pomiar światła (sprawdzisz, jak bardzo jest jasno lub ciemno).

Pomiar trzeba najpierw oprogramować i przetestować.

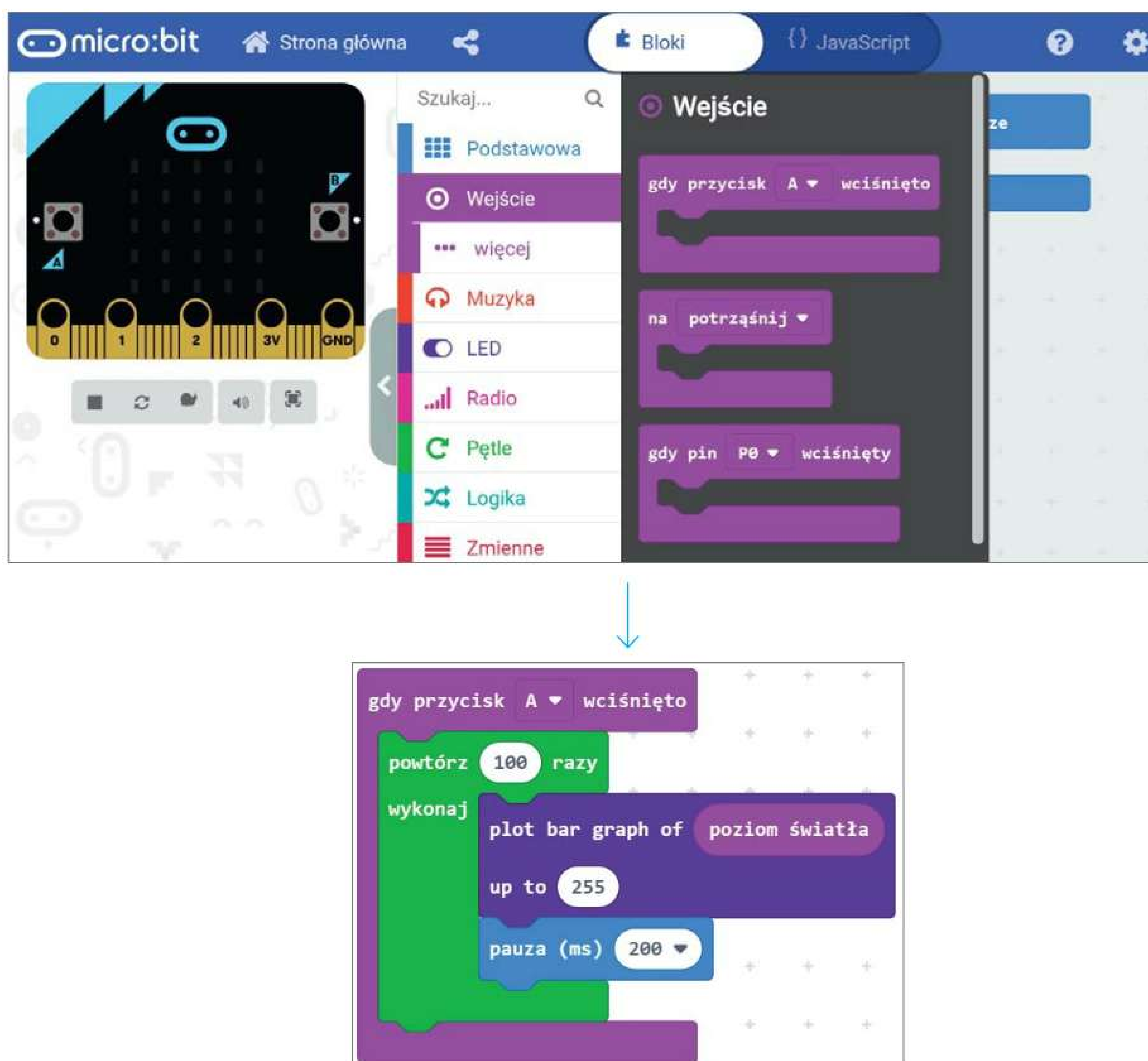
► Otwórz stronę **makecode.microbit.org** i kliknij przycisk **Nowy projekt**, aby otworzyć edytor blokowy. Nadaj projektowi nazwę, np.: *Pomiar światła*. Zauważ, że w opcjach kodu możesz wybrać bloki lub języki JavaScript i Python.



PRZÓD PŁYTKI

1. dwa programowalne przyciski (uproszczona klawiatura)
2. matryca LED 5×5 z czerwonymi diodami (uproszczony ekran)
3. pięć pinów (złączy, za pomocą których można łączyć z płytką inne urządzenia)

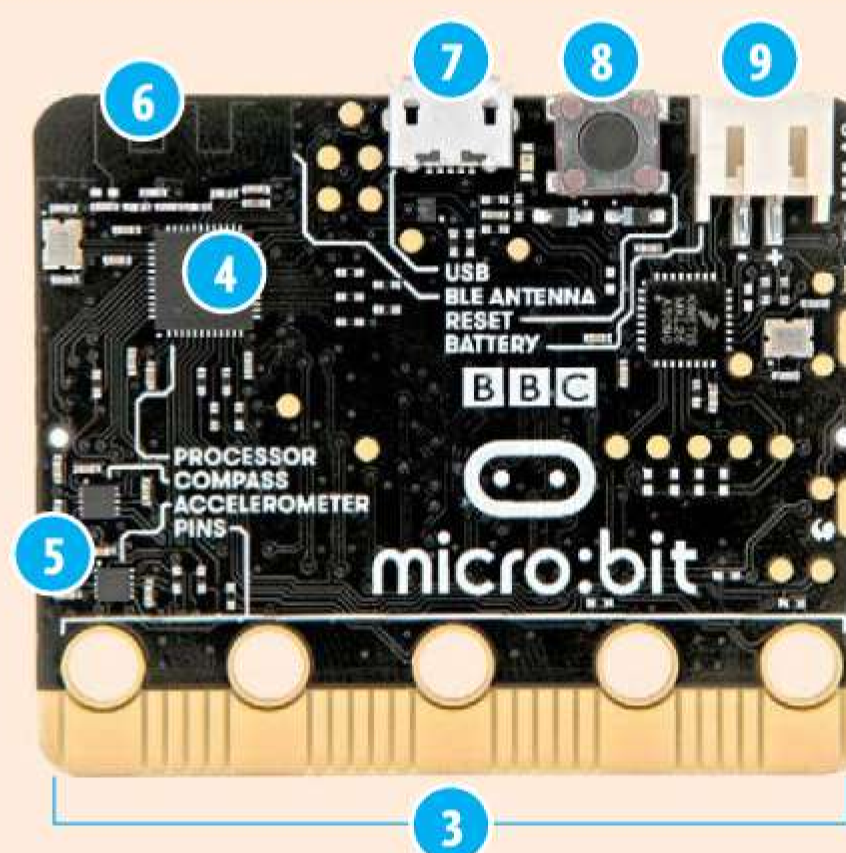
► Zbuduj skrypt – kliknij grupę bloków **Wejście** i przeciągnij do okna projektu blok `gdy przycisk A wciśnięto`, a następnie dołącz pozostałe bloki zgodnie z opisem pomiaru.



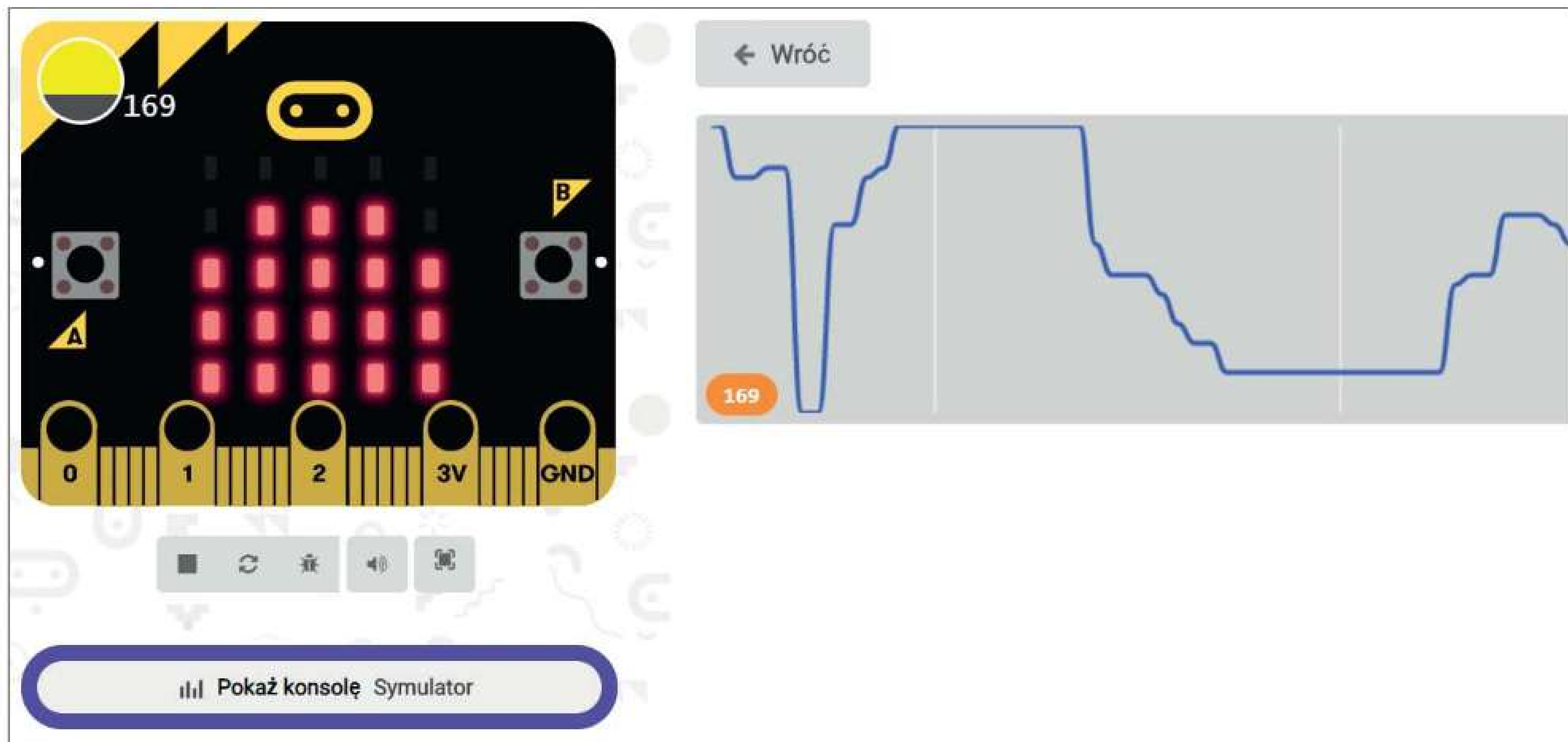
Rys. 1. Układanie bloków w oknie projektu

TYŁ PŁYTKI

- **3** pięć pinów
- **4** procesor zawierający pamięć i czujnik temperatury
- **5** czujniki pola magnetycznego i przyspieszenia
- **6** antena radiowa do połączeń bezprzewodowych z innymi urządzeniami
- **7** gniazdo USB
- **8** przycisk powodujący powrót urządzenia do stanu początkowego
- **9** gniazdo zasilania



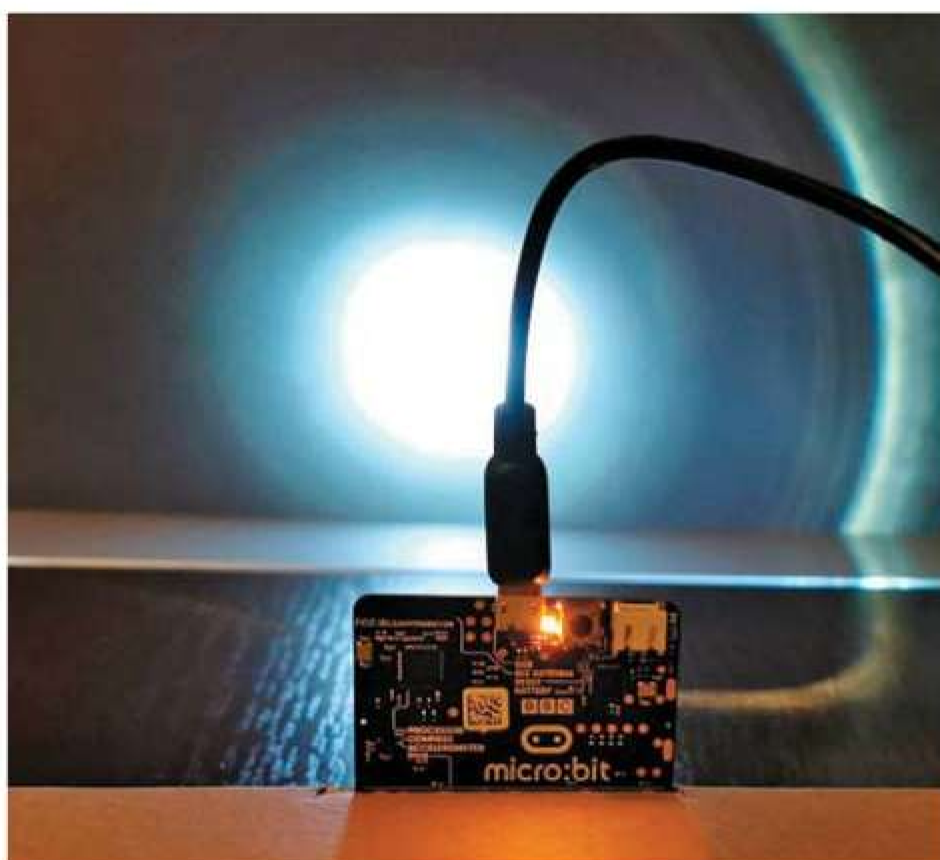
► Przetestuj działanie programu – w oknie symulatora kliknij kolejno przycisk **A** i przycisk **Pokaż konsolę Symulator**, a następnie za pomocą „słoneczka" w lewym górnym rogu symuluj zmianę oświetlenia.



Rys. 2. Symulacja pomiaru poziomu światła

Jeśli symulacja działa poprawnie, można podłączyć płytkę do komputera za pomocą kabla USB i zabrać się do prawdziwych pomiarów. Będzie do tego potrzebnych kilka kolorowych kartonów i latarka. Trzeba też zaciemnić pomieszczenie.

Jak przebiega pomiar? Płytka mierzy poziom światła wokół siebie za pomocą diod LED. Po wciśnięciu przycisku A płytka może np. stukrotnie wyświetlić diagram słupkowy poziomu światła aż do poziomu równego 255 (gdzie 0 oznacza ciemność, a 255 – jaskrawe światło), z przerwą 0,2 s między kolejnymi diagramami. Przykładowe ustawienie eksperymentu pokazano na poniższym zdjęciu.

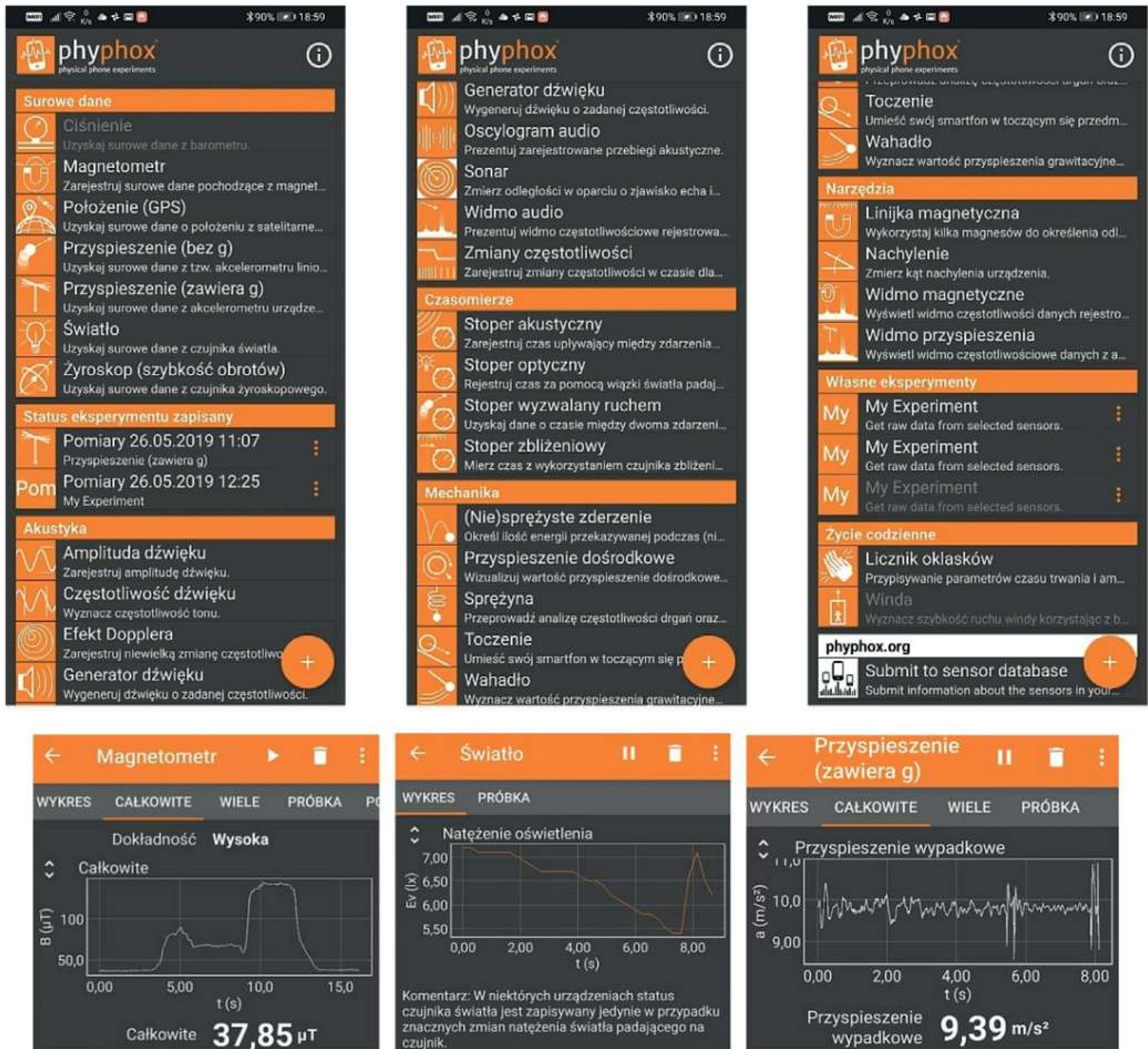


Rys. 3. Micro:bit w trakcie pomiaru

► Ustaw kolorowe kartony, włącz latarkę i do dzieła. Masz 100 * 200 ms, czyli 20 s.

POMIARY Z TELEFONEM

Ten sam pomiar można wykonać za pomocą telefonu z zainstalowaną bezpłatną aplikacją Phyphox, którą można pobrać w sklepach Google Play i AppStore. Aplikacja oferuje wiele możliwości. Jeśli któryś z opisów pomiaru jest szary, to znaczy, że telefon nie ma odpowiedniego miernika.



pomiar pola magnetycznego (zbliżanie nożyczek) | pomiar natężenia światła (zasłanianie kamery) | pomiar przyspieszenia ziemskiego

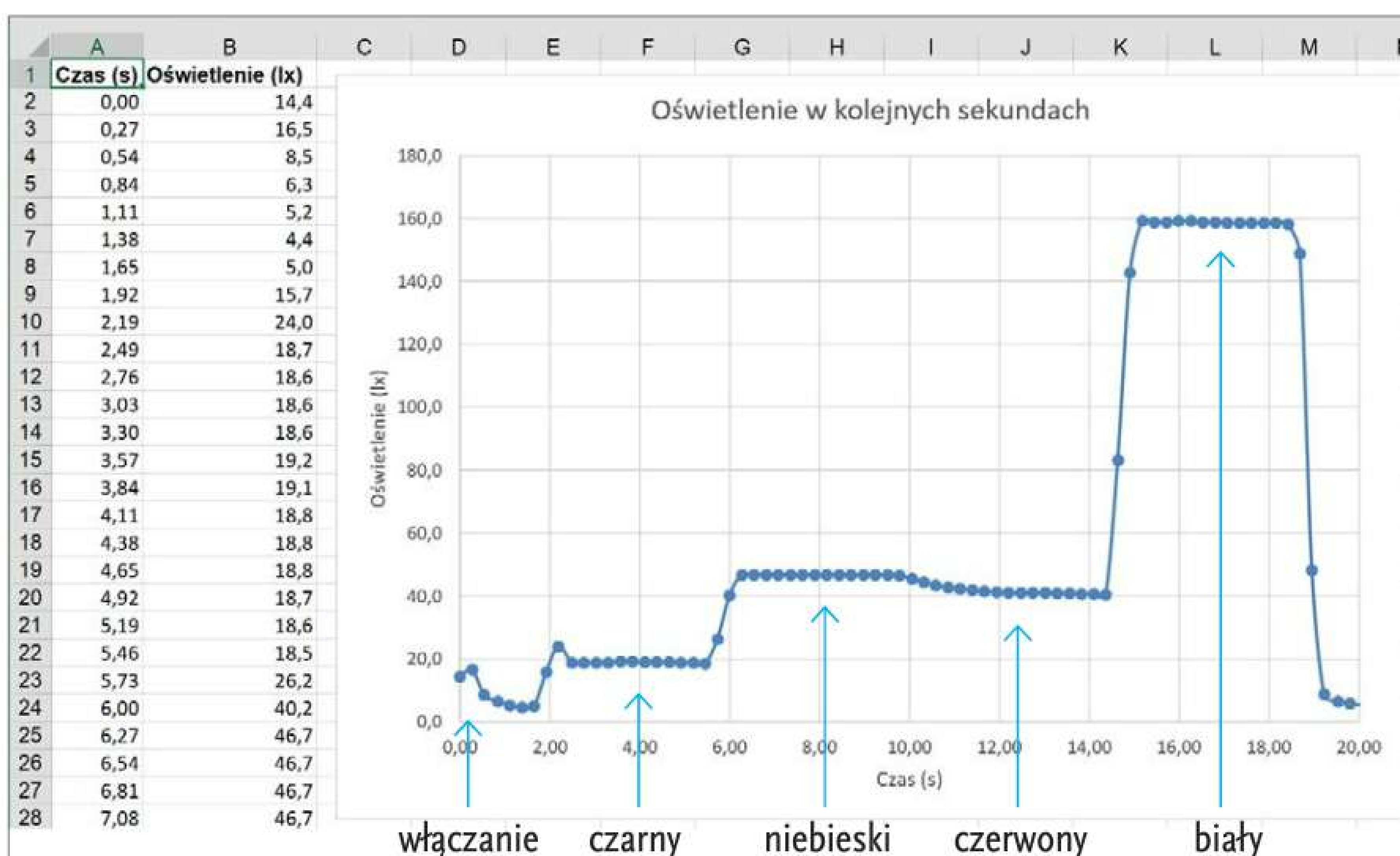
Rys. 4. Zestawienie pomiarów, które można wykonać w aplikacji Phyphox, i wyniki trzech przykładowych eksperymentów

- Ustaw kolorowe kartony i włącz latarkę.
- Wykonaj pomiar światła odbitego – wybierz w aplikacji eksperyment **Światło**, ustaw telefon ekranem w stronę kartonów i włącz pomiar.
- Zmieniaj kartony, a na końcu wyłącz pomiar.

- Wyeksportuj wyniki pomiarów do arkusza kalkulacyjnego, aby wykonać wykres punktowy – kliknij menu (trzy kropki w prawym górnym rogu), wybierz **Eksportuj dane**, jako format danych wybierz **Excel** i kliknij **OK**.

Dane możesz przesłać pocztą elektroniczną, zapisać na Dysku Google lub w pamięci swojego telefonu.

- Wczytaj zapisane dane na swoim komputerze.
- Otwórz plik danych w Excelu, zaznacz tabelę z pomiarami (**Ctrl+Shift+↓** – pomiarów jest zazwyczaj bardzo dużo) i wstaw wykres punktowy (**Wstawianie** → **Wykresy** → **Wstaw wykres punktowy (X, Y)**).



Czas (s)	Oświetlenie (lx)
0,00	14,4
0,27	16,5
0,54	8,5
0,84	6,3
1,11	5,2
1,38	4,4
1,65	5,0
1,92	15,7
2,19	24,0
2,49	18,7
2,76	18,6
3,03	18,6
3,30	18,6
3,57	19,2
3,84	19,1
4,11	18,8
4,38	18,8
4,65	18,8
4,92	18,7
5,19	18,6
5,46	18,5
5,73	26,2
6,00	40,2
6,27	46,7
6,54	46,7
6,81	46,7
7,08	46,7

Rys. 5. Przykładowe wyniki eksperymentu badającego natężenie światła w zależności od koloru kartonu, od którego światło się odbijało

ZADANIA

1. Wykorzystaj bloki **zawsze** oraz **pokaż LED** z grupy **Podstawowa** i utwórz program wyświetlający pulsujące serce.

2. Wykonaj pomiar pola magnetycznego w aplikacji Phyphox, wykorzystując **Magnetometr**. W trakcie pomiaru zbliż do telefonu żelazny przedmiot, np. nożyczki. Wyeksportuj wyniki pomiaru do arkusza i opracuj je w postaci wykresu punktowego.

3. Wybierz w aplikacji Phyphox pomiar położenia (GPS) i zbadaj jego możliwości.

4. Zapoznaj się z możliwościami pomiarowymi aplikacji Phyphox, a następnie zaprojektuj i wykonaj pomiar według własnego pomysłu. Opracuj wyniki eksperymentu w arkuszu kalkulacyjnym.

4.6 Podróże z komputerem

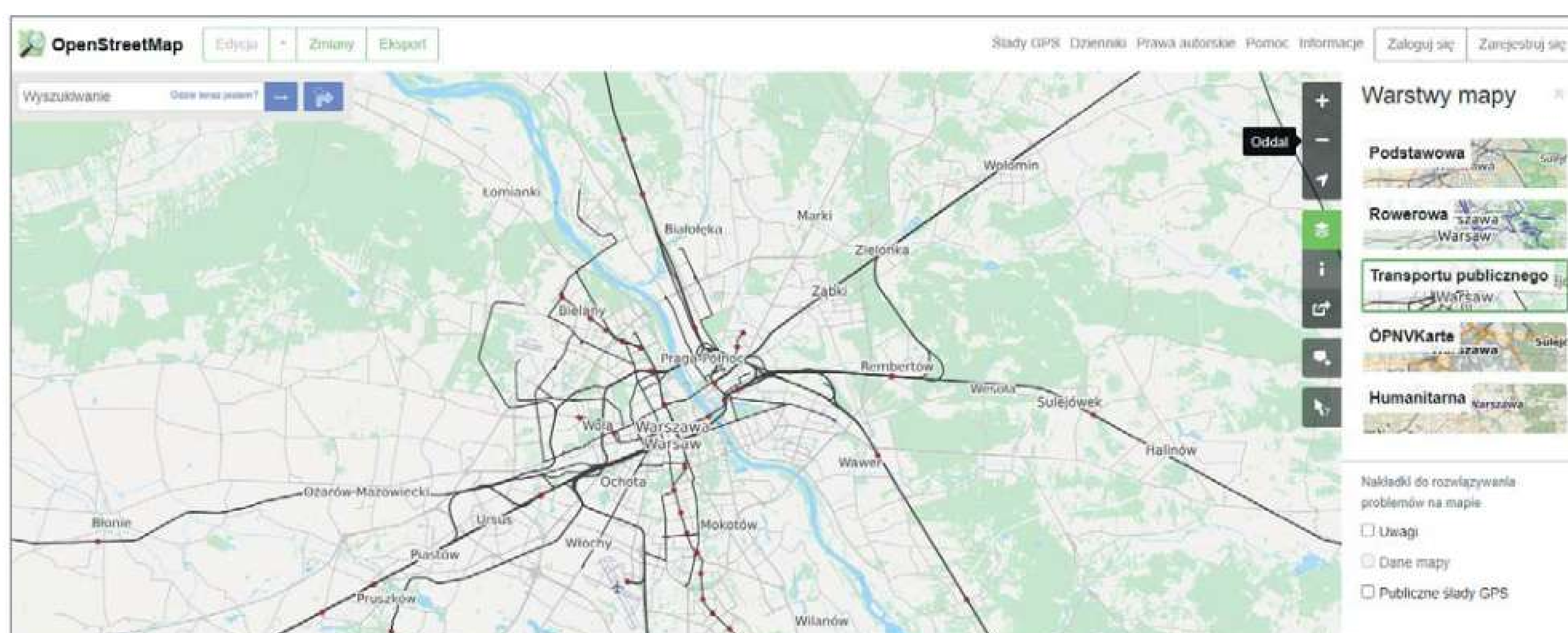
DOWIESZ SIĘ,

- z jakich serwisów zawierających mapy internetowe warto korzystać,
- czym są GIS i GPS,
- jak korzystać z aplikacji mapowych posługujących się GIS i GPS.

Planowanie podróży staje się coraz prostsze dzięki komputerom, nawigacji satelitarnej i systemowi informacji geograficznej. Mapy, traktowane jako przybliżone modele świata rzeczywistego, stały się bardzo przydatnym narzędziem. W internecie jest wiele serwisów przedstawiających mapy Polski i świata. Często zawierają szczegółowe plany miast oraz umożliwiają wybór drogi dojazdu.

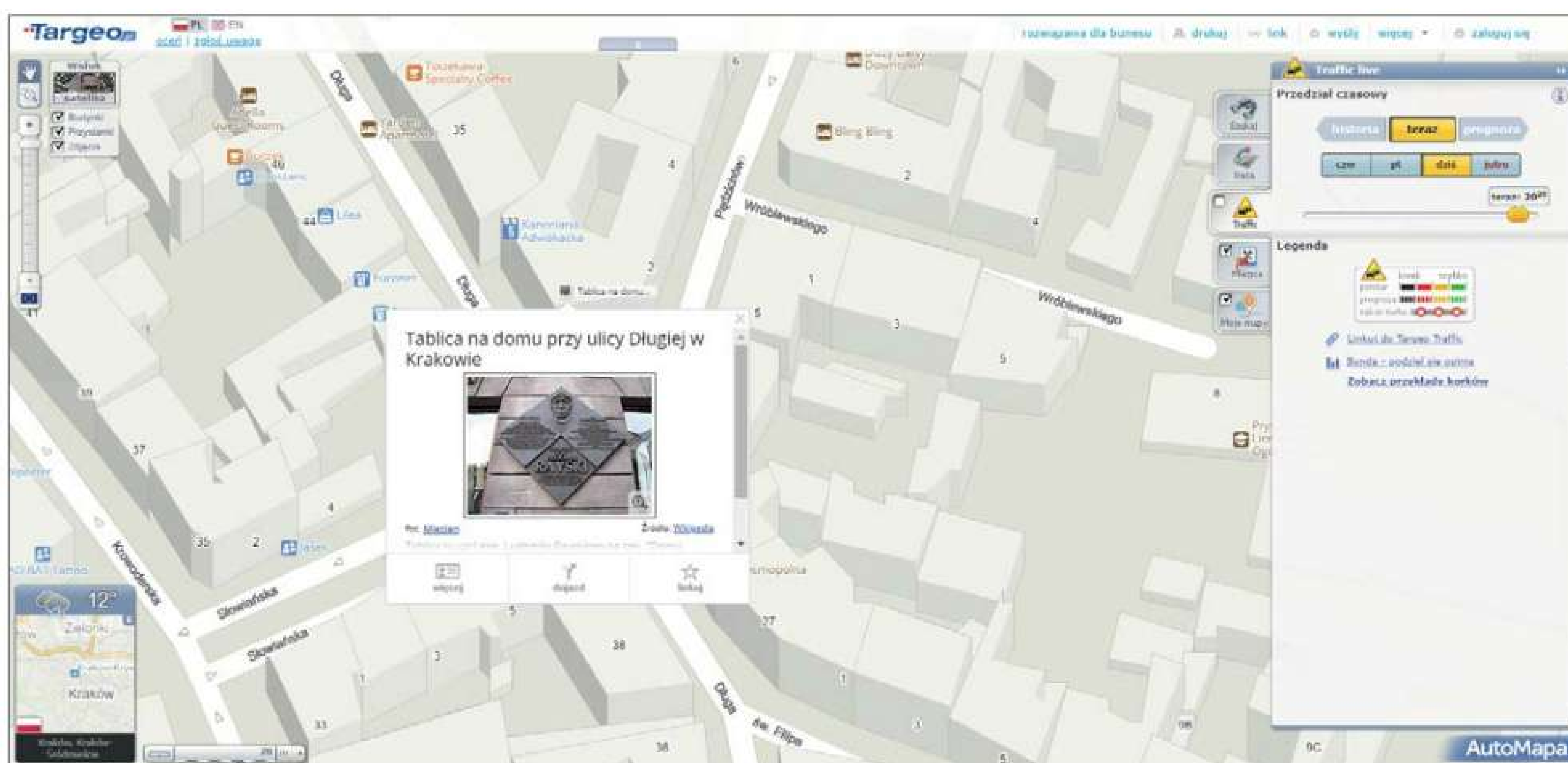
MAPY INTERNETOWE

Mapy i plany prezentowane przez OpenStreetMap (**www.openstreetmap.org**) wyposażono w warstwy: podstawową, rowerową, transportu publicznego (standardową i niestandardową ÖPNVKarte) oraz humanitarną (dla potrzeb służb ratunkowych). Poza tym mapa rejestruje ślady GPS, które można wczytać na własną mapę i korzystać z nich, o ile zostały udostępnione publicznie. Dokładna legenda ułatwia rozpoznawanie obiektów. Jest to projekt realizowany na otwartej licencji – mapę można edytować, dodając lub aktualizując informacje, np. o drogach czy kawiarniach, i dowolnie użytkować (wystarczy tylko zaznaczyć, że dane pochodzą z OpenStreetMap). Na podstawie danych z OpenStreetMap działa aplikacja OsmAnd, z której można korzystać w systemach Android oraz iOS.



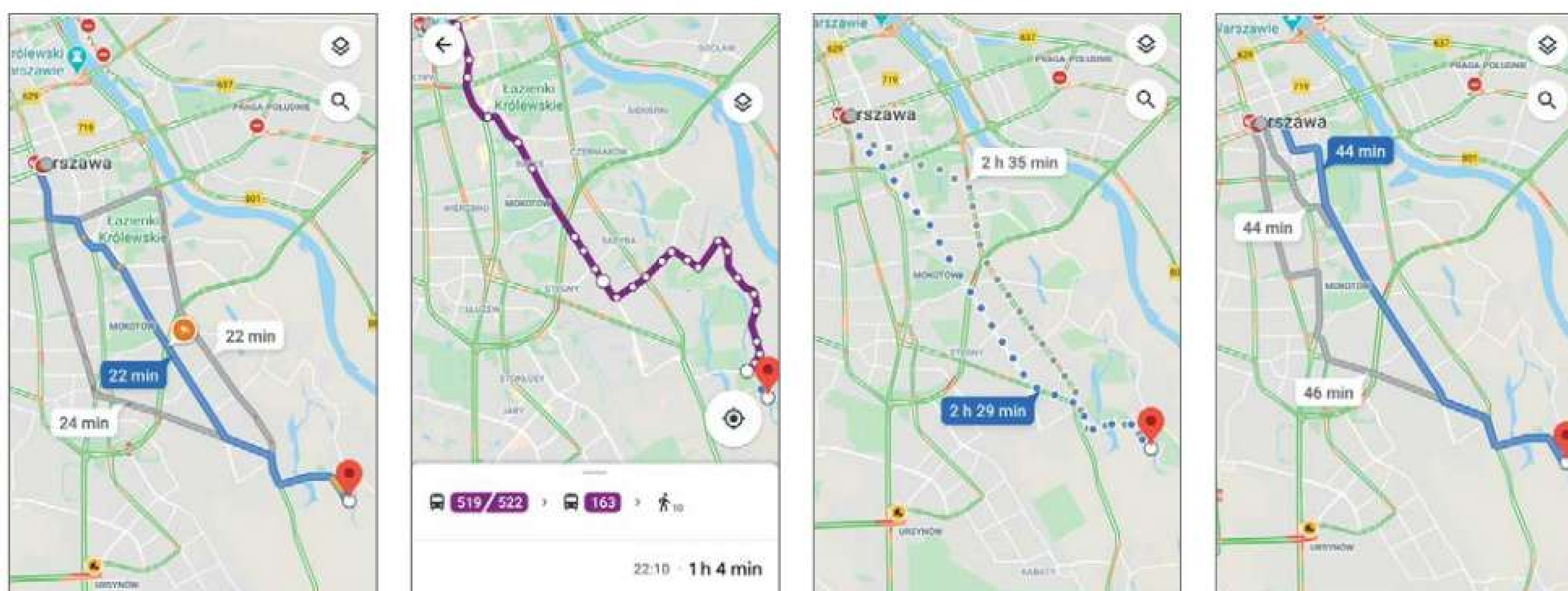
Rys. 1. Warszawa z warstwą transportu publicznego na OpenStreetMap

Mapa Targeo (**mapa.targeo.pl**) umożliwia wyszukiwanie adresów w wielu miastach Polski i państw europejskich, a także planowanie trasy przejazdu między dwoma miejscami na całym świecie. Działa na podstawie danych wykorzystywanych w programie do nawigacji satelitarnej Automapa. Jest wyposażona w mechanizm wbudowania fragmentu mapy we własną stronę internetową. Opracowano również wersję do zainstalowania na urządzeniach mobilnych. Oprócz bardzo dokładnej lokalizacji budynków wyświetla wybrane obiekty, pokazuje natężenie ruchu samochodowego, a także zdjęcia zamieszczone przez użytkowników.



Rys. 2. Fragment mapy Targeo ze zdjęciem zamieszczonym przez użytkownika

Serwis Mapy Google (**maps.google.com**) łączy mapy rysowane z mapami satelitarnymi. Dla wielu rejonów, np. dla Warszawy, zamieszczono w nim bardzo dokładne zdjęcia satelitarne. Serwis pozwala wyznaczać trasy dojazdu między dowolnymi dwoma miejscami na całym świecie, nawigować podczas jazdy samochodem i sprawdzać natężenie ruchu albo przebieg tras środków transportu publicznego czy rowerowych.

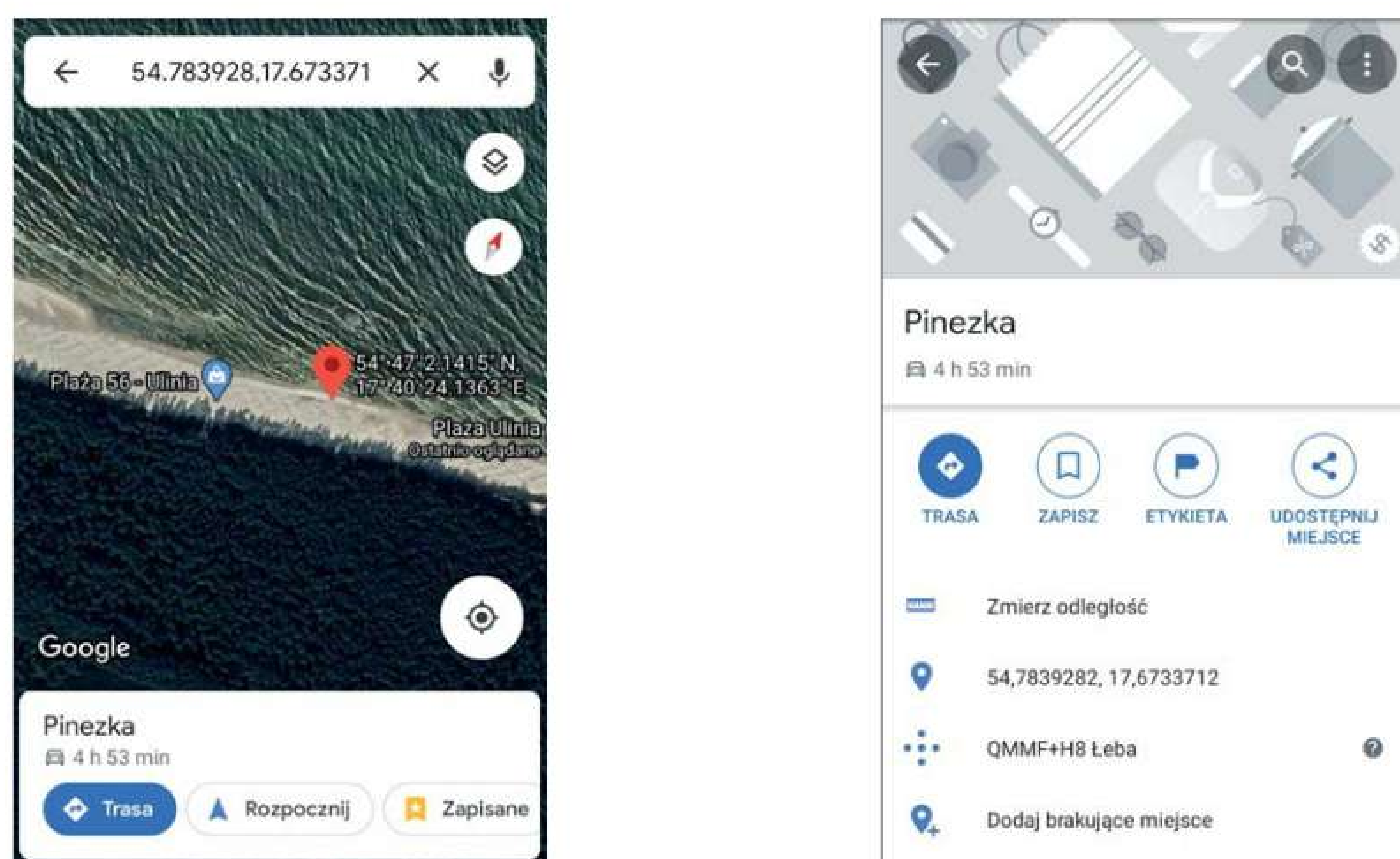


Rys. 3. Trasa z Dworca Centralnego w Warszawie do Wilanowa: samochodem, komunikacją miejską, pieszo i rowerem

Poza tym po odpowiednim powiększeniu fragmentu mapy lub zdjęcia automatycznie uruchamia się funkcja **Street View** (o ile jest w tym miejscu dostępna), która pozwala wirtualnie podróżować po okolicy, a także obejrzeć panoramiczne zdjęcia konkretnego miejsca. Miejsca, gdzie działa **Street View**, można zobaczyć po przeciągnięciu na obszar mapy ikony przedstawiającej żółtego ludzika, znajdującej się w prawym dolnym rogu mapy, obok obszaru skalowania. Mapy Google można też zainstalować na urządzeniach mobilnych. Tam są najbardziej przydatne na co dzień.

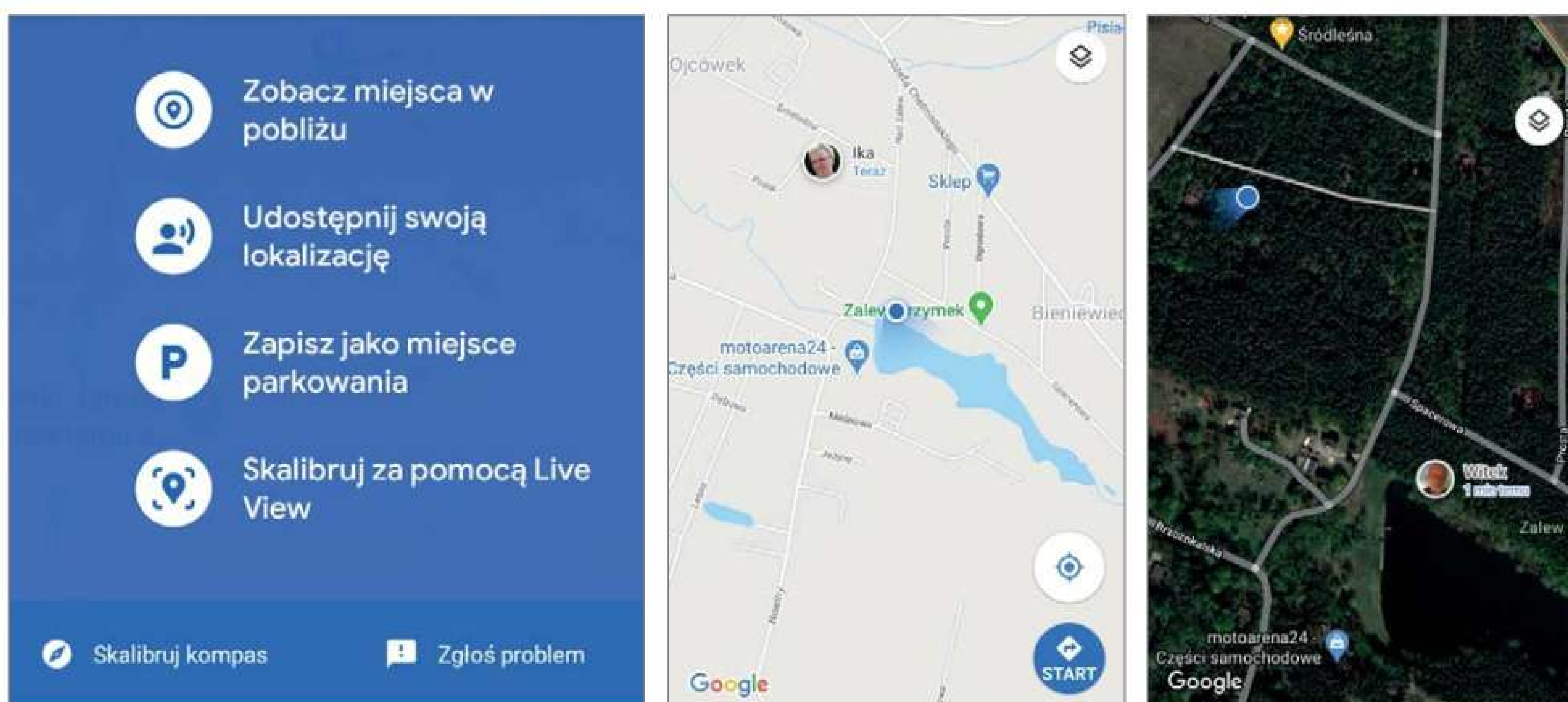
Menu Map Google umożliwia też osadzenie mapy na innej stronie internetowej bądź przekazanie linków do wybranych fragmentów map, dodawanie miejsc, zgłaszanie przeszkód na drodze (wypadków, zagrożeń).

Co więcej, gdy zaprasza się gości, można przesłać im zamiast adresu fragment mapy. Będą mogli sami zaplanować trasę dojazdu lub dojścia do ustalonego miejsca. Wystarczy na urządzeniu mobilnym zaznaczyć miejsce pinezką (dotknięciem palca) na mapie, a następnie udostępnić za pomocą przycisku **Udostępnij miejsce**. Link można przekazać e-mailem, esemesem lub innym sposobem komunikacji dostępnym z telefonu czy komputera. Zawiera on nie tylko obraz miejsca i położenie na mapie, lecz także współrzędne geograficzne i inne dostępne dla tego miejsca szczegółowe informacje.



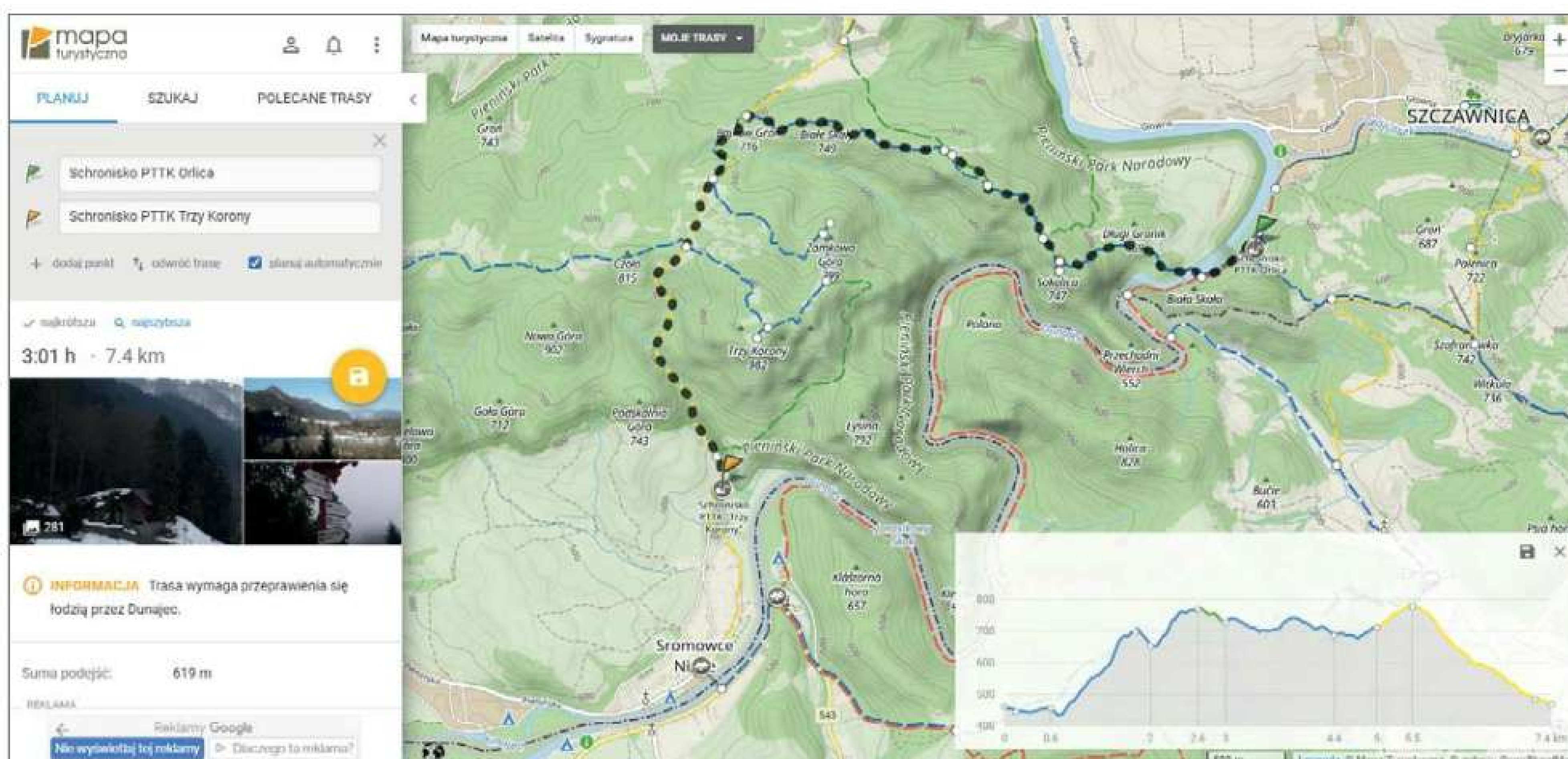
Rys. 4. Przesyłanie linku do miejsca na mapie

Jeszcze inną możliwość daje udostępnienie swojego położenia w czasie rzeczywistym. Trzeba dotknąć palcem niebieskiej kropki obrazującej na mapie nasze położenie, wybrać z opcji **Udostępnij swoją lokalizację**, a następnie wskazać osobę z listy odbiorców. Jeśli odbiorca będzie miał otwarte Mapy Google, zobaczy przesłaną lokalizację. Może też kliknąć przesłany link i aplikacja sama się otworzy. Standardowo udostępnianie trwa godzinę i po tym czasie trzeba je ponowić, jeśli jest to potrzebne. Można również ustawić inny czas jego trwania w opcjach udostępniania.



Rys. 5. Udostępnienie własnej lokalizacji i widok wzajemnego równoczesnego udostępnienia na dwóch telefonach

A jeśli z telefonem komórkowym bądź komputerem wyjedziesz na wakacje, możesz zaplanować nie tylko dojazdy, lecz także trasy wędrówek. Serwis Mapa turystyczna (**mapa-turystyczna.pl**) to odpowiednik przewodnika turystycznego z zaznaczonymi szlakami i możliwością planowania tras. Dzięki osadzeniu w Mapach Google ma bardzo podobne zasady działania i możliwość udostępniania fragmentów map. Posługuje się też widokiem satelitarnym i klasyczną mapą turystyczną.



Rys. 6. Trasa zaplanowana i wyznaczona w serwisie Mapa turystyczna

GIS I GPS

Działanie serwisów mapowych opiera się na systemie informacji geograficznej GIS (akronim od ang. *Geographic Information System*; czytaj: dżijegrafyk infemejszyn system). Powstał on dzięki połączeniu narzędzi informatycznych z trzech dziedzin: geografii, kartografii i geodezji. Bazy danych tego systemu łączą dane dotyczące lokalizacji miejsca na Ziemi z danymi opisowymi o obiektach, które znajdują się w konkretnym

miejscu. Istotnym elementem jest również rozbudowany sposób wizualizacji – przedstawiania miejsc i obiektów na ekranie. Bardzo rozbudowany system informacji oraz wizualizacji zawiera znany ci już program Google Earth. Dowiesz się o nim więcej z lekcji 5.2.

GIS jest coraz częściej wykorzystywany w samochodach wraz z systemem nawigacji satelitarnej GPS (akronim od ang. *Global Positioning System*; czytaj: głołbyl pozyszynyn system). Dzięki połączeniu z systemem satelitów urządzenie bardzo dokładnie lokalizuje swoje położenie na powierzchni Ziemi i prowadzi kierowcę pod wybrany adres, cały czas pokazując na ekranie aktualną pozycję pojazdu. Może też informować o ruchu drogowym, korkach, remontach dróg, objazdach – wszystko zależy od tego, do jakich danych oprogramowanie ma dostęp.

Z GIS i GPS korzystają różnorodne aplikacje: krokomierze, aplikacje monitorujące aktywność sportową, aplikacje pomiarowe. Aplikacja Phyphox, wykorzystywana podczas poprzedniej lekcji, może – za pomocą GPS – monitorować przemieszczanie się jej użytkownika nawet na zwykłym spacerze. Wystarczy włączyć opcję **Położenie GPS**, a po zakończeniu pomiaru wywołać opcję **Próbka**, aby wyświetlić pełne podsumowanie.

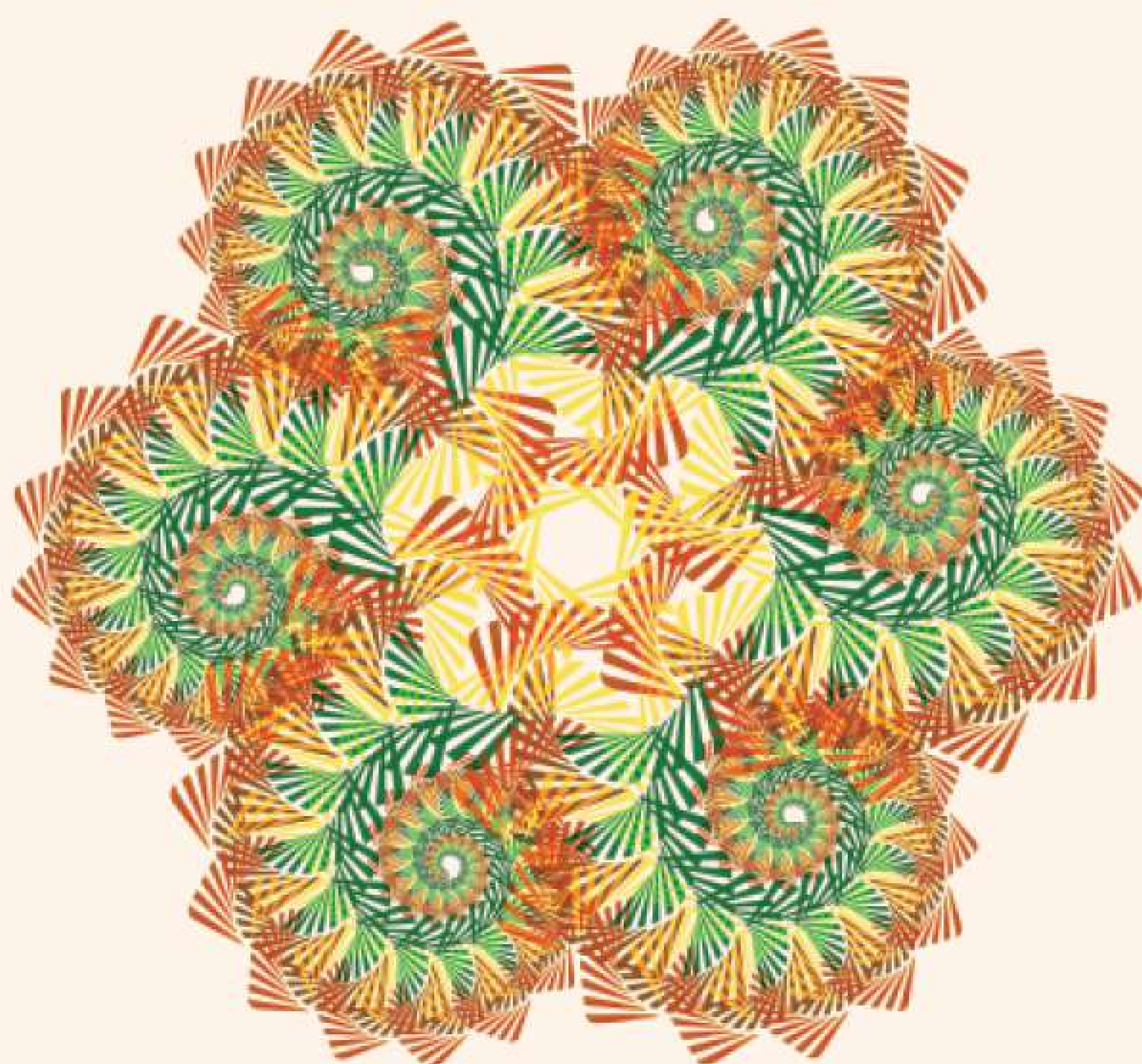
ZADANIA

1. Zaplanuj podróż lub wycieczkę szkolną. Znajdź najlepszą trasę dojazdu i określ czas podróży. Wyszukaj informacje o możliwych postojach, interesujących miejscach, które warto zwiedzić, zaplanuj trasy wędrówek. Porównaj wyniki uzyskane w dwóch różnych serwisach z mapami. Zwróć uwagę na interfejs użytkownika, sposoby wizualizacji oraz różnorodność danych, które można uzyskać. Przedyskutuj swoje spostrzeżenia z koleżankami i kolegami.
2. Podczas jazdy samochodem lub autobusem włącz na smartfonie serwis Mapy Google i prześledź, jakie informacje i możliwości oferuje przy planowaniu tras dojazdu oraz wyszukiwaniu potrzebnych informacji lub komunikatów o bieżących zagrożeniach. Wyłącz głos, aby nie przeszkadzać kierowcy lub współpasażerom.
3. Wykonaj monitorowanie przemieszczania się w aplikacji Phyphox i zanotuj opis spaceru lub wycieczki: opisz trasę oraz obiekty i uzupełnij informacje o pokonanym dystansie, zmianie położenia i prędkości przemieszczania się.



Jak przeprowadzać symulacje zdarzeń losowych, gdy bezpośrednie doświadczenia nie są możliwe?

Symulacja zdarzeń losowych jest możliwa np. w arkuszu kalkulacyjnym. Pozwala na prowadzenie doświadczeń losowych na skalę niedostępną w rzeczywistości. Liczby pseudolosowe w wystarczającym stopniu przybliżają zjawiska losowe zachodzące w naturze.



Jak zaprogramować rysowanie fraktali?

Ponieważ fraktale charakteryzuje samopodobieństwo (małe fragmenty figury są podobne do całej figury), do programowania można wykorzystać metodę rekurencji – czy to poprzez układanie skryptu z gotowych bloków (np. w Scratchu), czy to poprzez wpisywanie poleceń (np. w Pythonie lub JavaScripcie).

Czy mapa jest modelem?

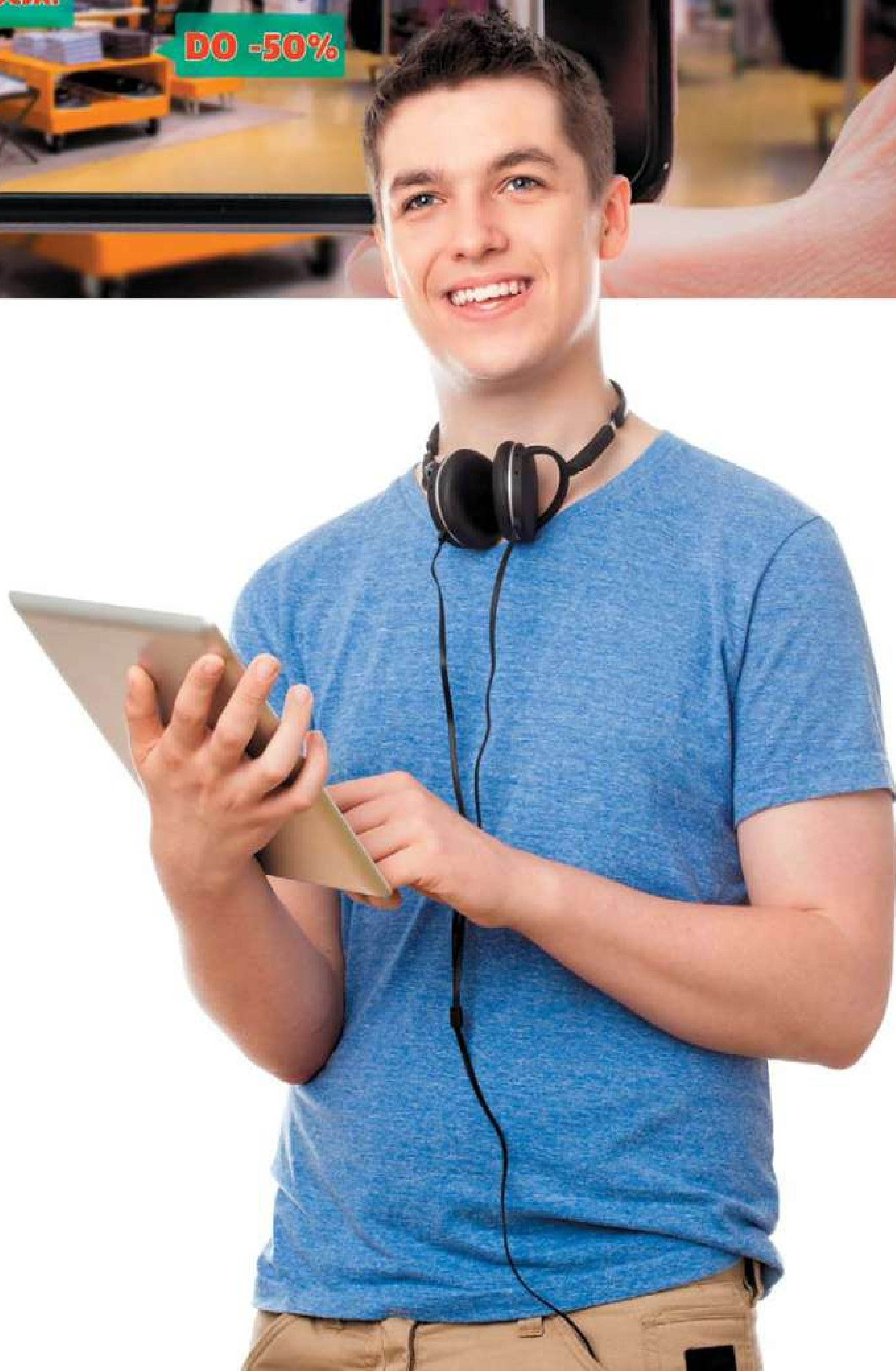
Korzystanie z aplikacji mapowych i turystycznych pozwala zaplanować podróż ze wszystkimi niemal szczegółami. Modele obiektów rzeczywistych umieszczone na warstwach mapy (np. budynki, drogi, sklepy, komunikacja, obiekty turystyczne) pozwalają odwzorować realny świat (stworzyć jego przybliżony model) w stopniu umożliwiającym planowanie dla własnych celów.



5 Lekcje w sieci



W 2009 r. powstało urządzenie wykorzystujące technologię szóstego zmysłu, które odczytuje zaprogramowane wcześniej gesty, przetwarza je i odpowiednio na nie reaguje, np. wyświetla szukane informacje albo tarczę zegarka lub klawiatury telefonu na dowolnej powierzchni. Więcej o tej technologii dowiesz się na stronie: **www.ted.com/talks/pranav_mistry_the_thrilling_potential_of_sixthsense_technology**.



5.1 Rozwijaj zainteresowania w sieci

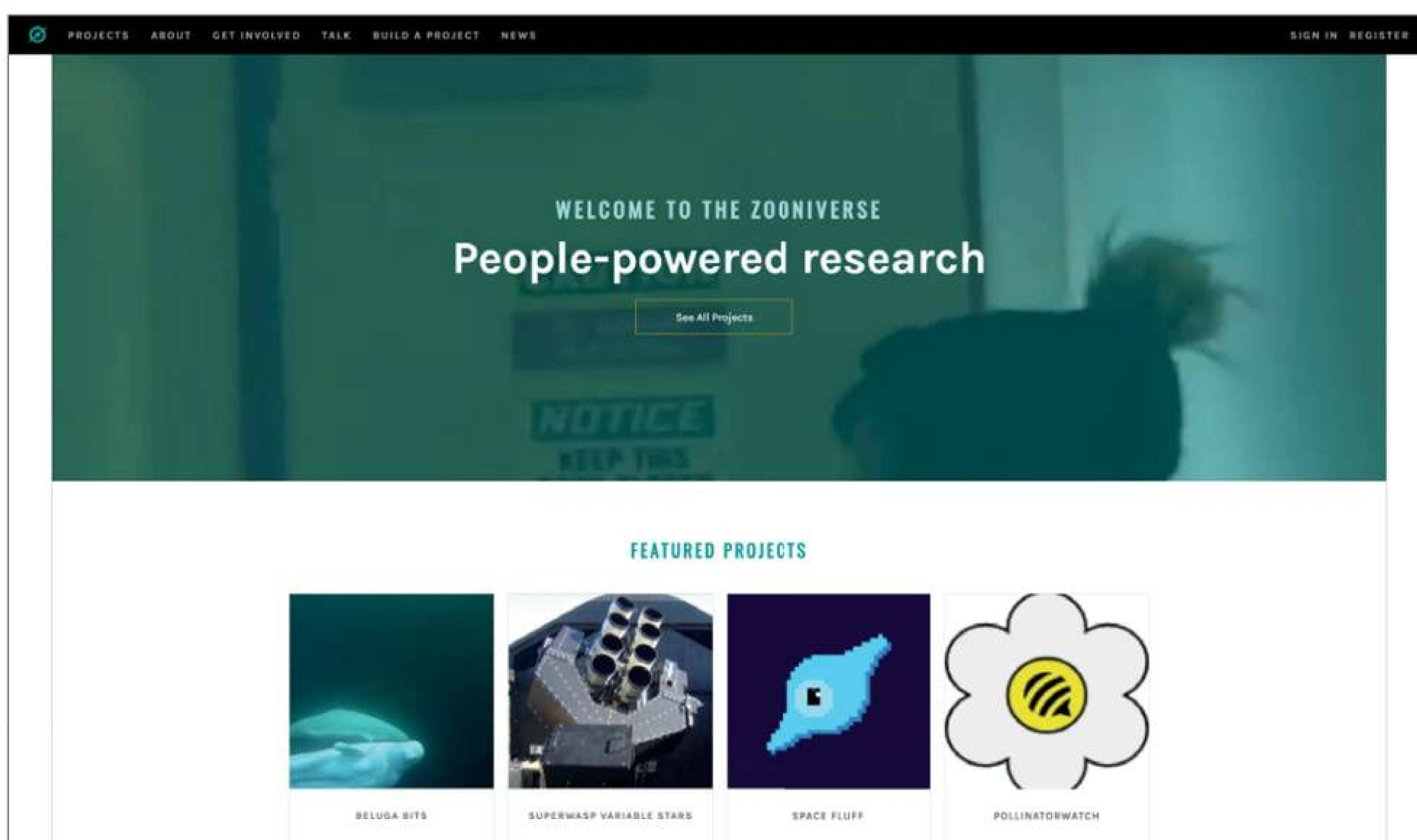
DOWIESZ SIĘ,

- w jakie projekty o charakterze naukowym może zaangażować się każdy internauta,
- jakie są możliwości rozwijania zainteresowań w sieci.

Podczas tej lekcji poznasz serwisy, które umożliwiają rozwijanie umiejętności, wiedzy i zainteresowań w sieci oraz udział w ciekawych przedsięwzięciach naukowych.

ZOONIVERSE

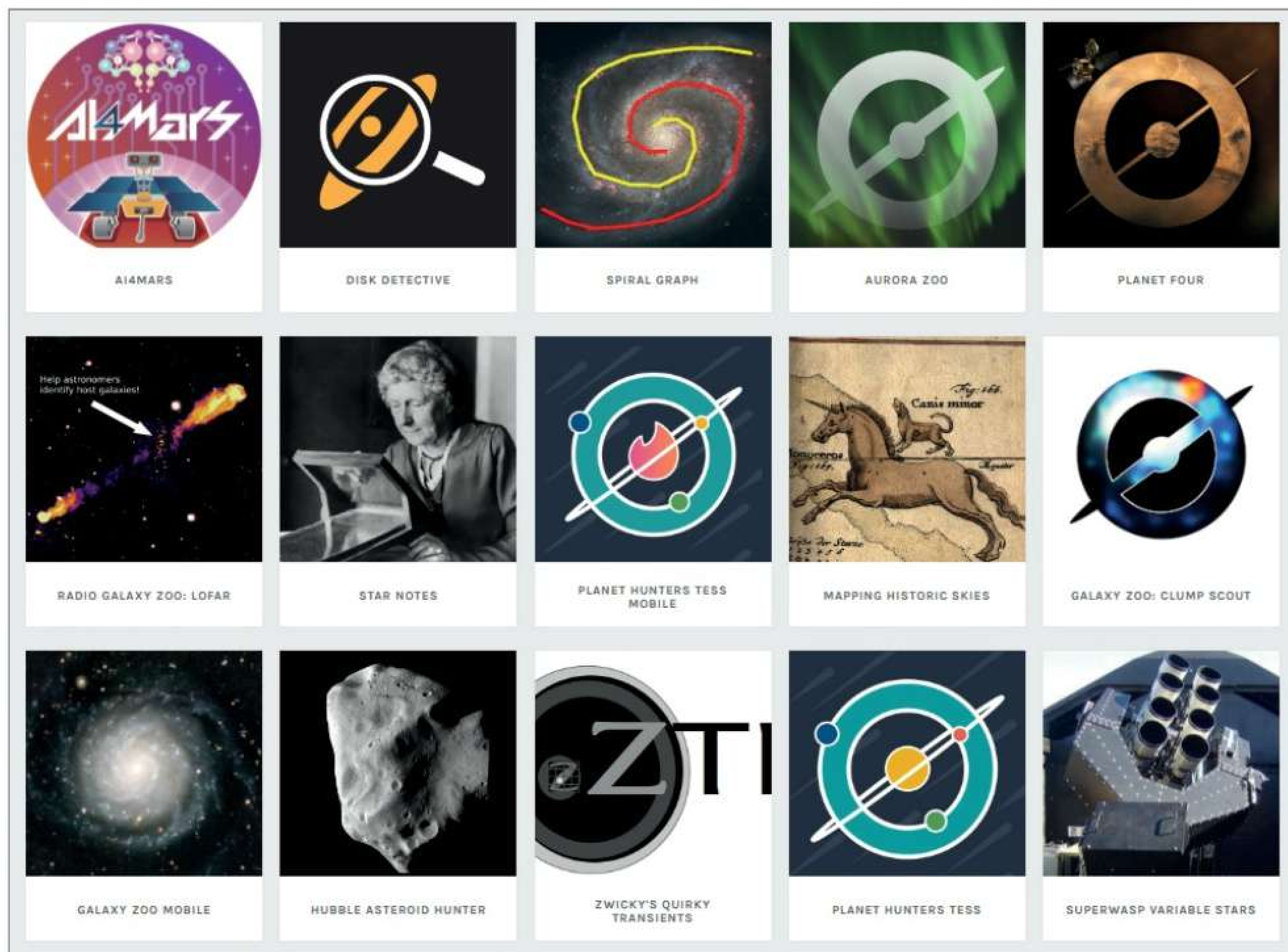
Na platformie **www.zooniverse.org** (a także w aplikacji mobilnej o tej samej nazwie) zgromadzono naukowe projekty badawcze, w które może się zaangażować każdy chętny internauta. W odróżnieniu od znanych ci z lekcji 3.5 projektów przetwarzania rozproszonego wymagają one od uczestników aktywnego udziału. Naukowcy dostarczają bazę obrazów do analizy, np. zdjęcia różnych ciał niebieskich, i opracowują metodę ich klasyfikowania oraz przewodnik, jak wykonywać analizę. Po zalogowaniu się na stronie projektu zarejestrowani wolontariusze, określani mianem „zooites", najpierw uczą się metody, a następnie dostają do klasyfikowania kolejne obrazy. Projektów jest coraz więcej, dotyczą m.in. sztuk pięknych, biologii, klimatologii oraz astronomii.



Rys. 1. Fragment strony z aktywnymi projektami na platformie Zooniverse.org

Oto przykładowe przedsięwzięcia z zakresu astronomii na platformie Zooniverse.org.

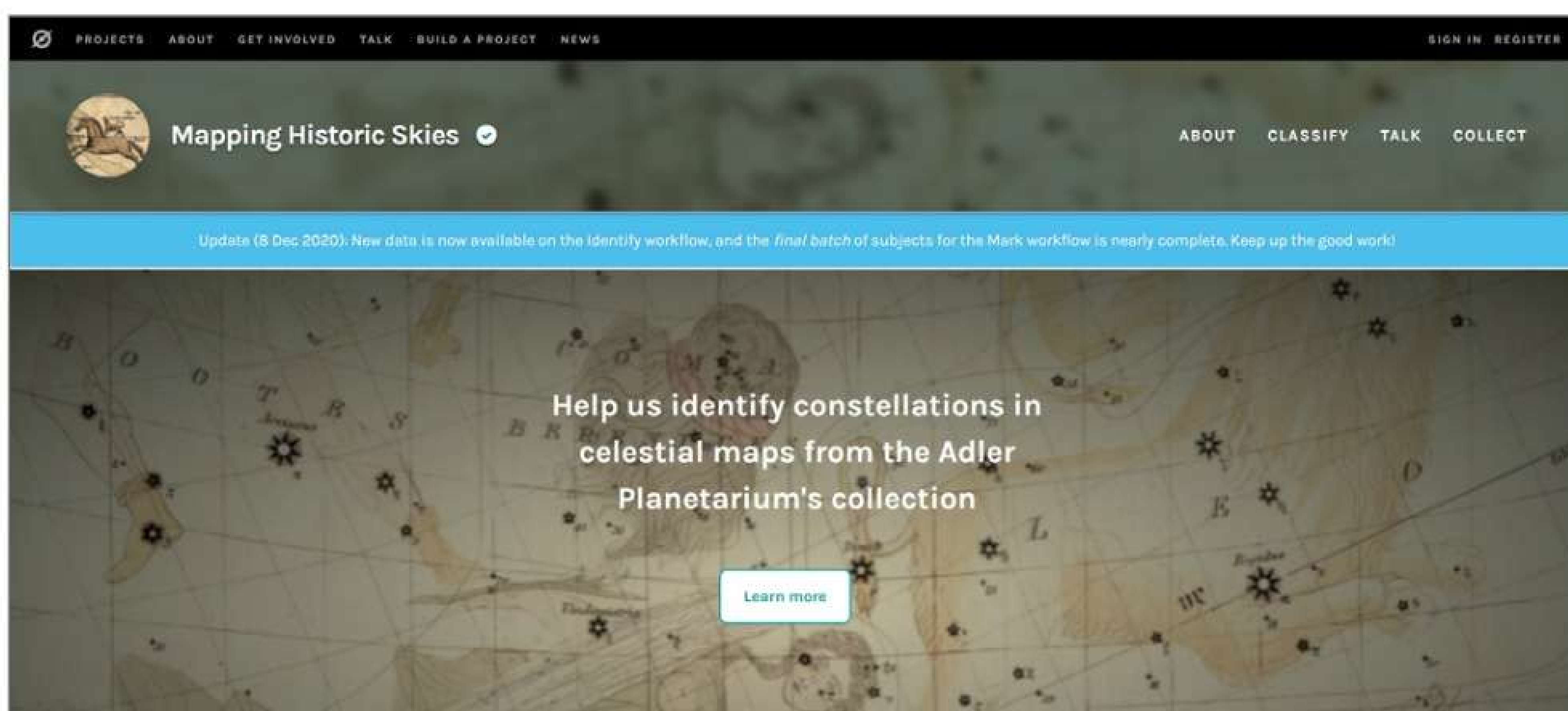
- **Galaxy Zoo** (czytaj: galaksi zuu; galaktyczne zoo) – badania odległych galaktyk, których być może jeszcze nikt nie oglądał; sukces uruchomionego w 2007 r. projektu przyczynił się do założenia w 2009 r. platformy Zooniverse.org.
- **Solar Stormwatch II** (czytaj: sołla stormłocz; obserwowanie burz słonecznych) – poszukiwanie plam, burz i innych zjawisk na powierzchni Słońca.
- **Planet Hunters** (czytaj: planet hanters; łowcy planet) – poszukiwanie planet krążących wokół innych gwiazd.
- **Milky Way Project** (czytaj: mylki łej prodżekt; projekt Droga Mleczna) – badanie obrazów naszej galaktyki w podczerwieni w poszukiwaniu interesujących obiektów.
- **Planet Four** (czytaj: planet for; czwarta planeta) – badanie zjawisk występujących w atmosferze Marsa.
- **Radio Galaxy Zoo** (czytaj: rejdjoł galaksi zuu; radiowo-galaktyczne zoo) – porównywanie obrazów podczerwonych i radiowych w poszukiwaniu czarnych dziur.
- **Backyard Worlds: Planet 9** (czytaj: bekjard łorlds: planet najn) – poszukiwanie obiektów w bliskim sąsiedztwie Układu Słonecznego; podczas przeglądania zdjęć z misji NASA WISE uczestnicy mogą znaleźć dziewiątą planetę albo gwiazdę bliższą niż Proxima Centauri (najbliższa z obecnie znanych gwiazd).

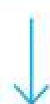


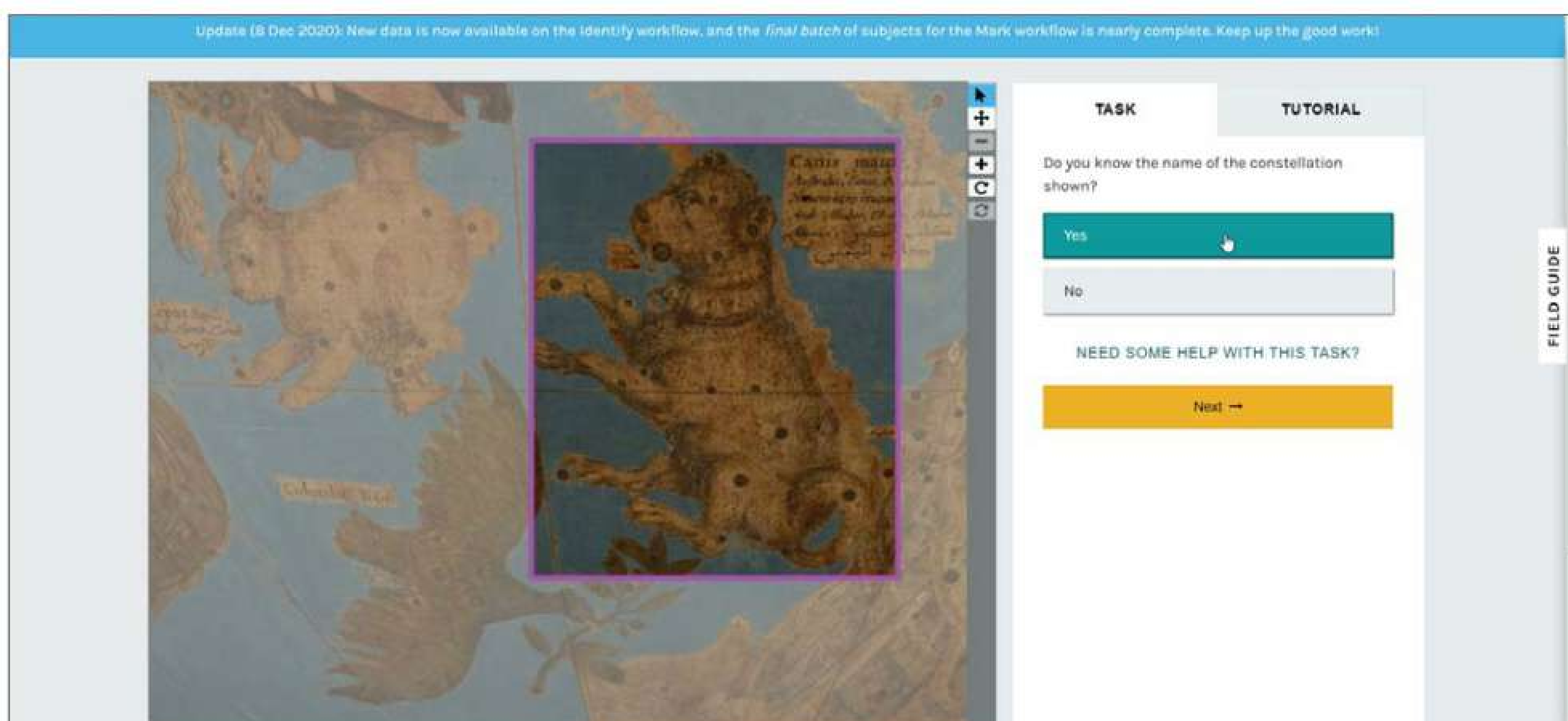
Rys. 2. Fragment listy projektów z kategorii **Space** (czytaj: spejs; kosmos)

Oto warte uwagi projekty z innych kategorii tematycznych.

- **Chimp & See** (czytaj: czimp end si) – sprawdzanie, czy zachowanie szympansów może nam powiedzieć więcej o pochodzeniu ludzkości. Identyfikowanie grup szympansów z różnych siedlisk i klasyfikowanie zachowań zwierząt na filmach z fotopułapek w 15 krajach Afryki. Klasyfikowanie zachowań, relacji i elementów środowiska pozwala ekstrapolować nowe pomysły na temat pochodzenia człowieka.
- **Ancient Lives** (czytaj: ejnszynt lajwz) – odczytywanie papirusów z Oksyrynchos. Uczestnicy litera po literze odcyfrowywali zapis widoczny na fotografii – na podstawie wzornika greckiego alfabetu z tamtych czasów. Projekt został zawieszony.
- **Mapping Historic Skies** (czytaj: maping historik skajz) – identyfikowanie konstelacji na mapach nieba z kolekcji Adler Planetarium. Konstelacje mogą przybierać różne formy, od zbioru kropek po szczegółową ilustrację, i przedstawiać wszystko, od postaci takich jak Orion, po zwierzęta i przedmioty, takie jak Wielki Wóz.







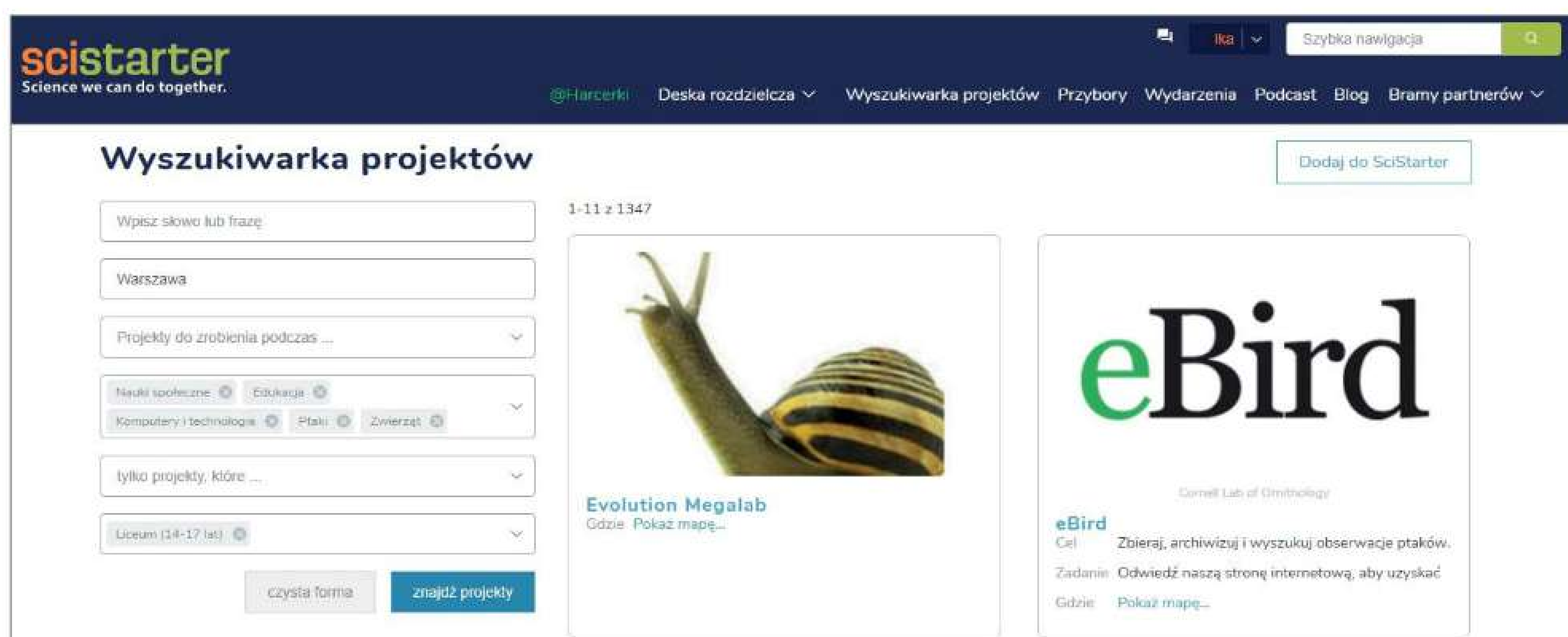
Rys. 3. Fragment strony głównej projektu Mapping Historic Skies oraz jedno z zadań

- **Earthquake Detective** (czytaj: erfkłejk detektiw) – klasyfikacja rodzajów zdarzeń sejsmicznych, takich jak trzęsienia ziemi lub drżenie, które są mierzone jako ruch powierzchni ziemi. Nie można usłyszeć surowych fal sejsmicznych, dlatego w tym projekcie zmieniono ich częstotliwości na słyszalne tony – wolontariusze dokonują klasyfikacji zdarzeń, po prostu ich słuchając.
- **Cheetahs of Central Namibia** (czytaj: czitaz of sentral namibia) – poszukiwanie gepardów w środkowej Namibii. Uczestnicy przeglądają zdjęcia z ok. 40 kamer i klasyfikują je w celu odnalezienia gepardów i zwierząt, które mogą być ich pożywieniem.
- **Snapshots at Sea** (czytaj: snapszots et si) – poszukiwanie wielorybów i delfinów. Projekt wykorzystuje m.in. zdjęcia z serwisu **www.happywhale.com**. Uczestnicy przeglądają zdjęcia w poszukiwaniu rzadkich okazów wielorybów i delfinów.

SCISTARTER

Najwięcej projektów tzw. nauki obywatelskiej skupia platforma **scistarter.org**, prowadzona przez Uniwersytet Stanowy Arizony. Nauka obywatelska jest określana jako współpraca między naukowcami a tymi z nas, którzy są po prostu ciekawi, zainteresowani i zmotywowani do zmiany. Wolontariusze dokumentują zmiany w przyrodzie (np. robią zdjęcia chmur lub strumieni), monitorują jakość wody i powietrza za pomocą czujników w smartfonach albo biorą udział w grach, przyczyniając się w ten sposób do postępów w badaniach w dziedzinie zdrowia i medycyny. Wiele ogólnych informacji przedstawiających naukę obywatelską można znaleźć na stronie **www.citizenscience.org**.

Na platformie SciStarter znajduje się ponad 1300 projektów. Aby łatwiej znaleźć interesujące zagadnienia, należy użyć wyszukiwarki Project Finder. Warto włączyć też automatyczne tłumaczenie strony.



Rys. 4. Wyszukiwanie projektów po włączeniu automatycznego tłumaczenia

Przykładowy projekt **Globe at Night** bada widoczność gwiazd w różnych miejscach na Ziemi, aby określić, jakie jest „zanieczyszczenie" światłem. Co ma zrobić uczestnik? W jasne i bezksiężycowe noce podczas 10-dniowej akcji trzeba wyjść na zewnątrz co najmniej godzinę po zachodzie słońca, ale przed 22 czasu lokalnego. Nie powinno

się stać w pobliżu źródła światła. Po 10 minutach, kiedy oczy przyzwyczają się do nocnego nieba, można przystąpić do realizacji zadania. Najpierw należy zlokalizować gwiazdozbiór miesiąca. W lecie będą to gwiazdozbiory Wolarza, Herkulesa i Łabędzia. Potem wystarczy porównać wygląd gwiazdozbioru z siedmioma rysunkami tego gwiazdozbioru i wybrać najlepiej pasujący.



Rys. 5. Fragment głównej strony projektu Globe at Night

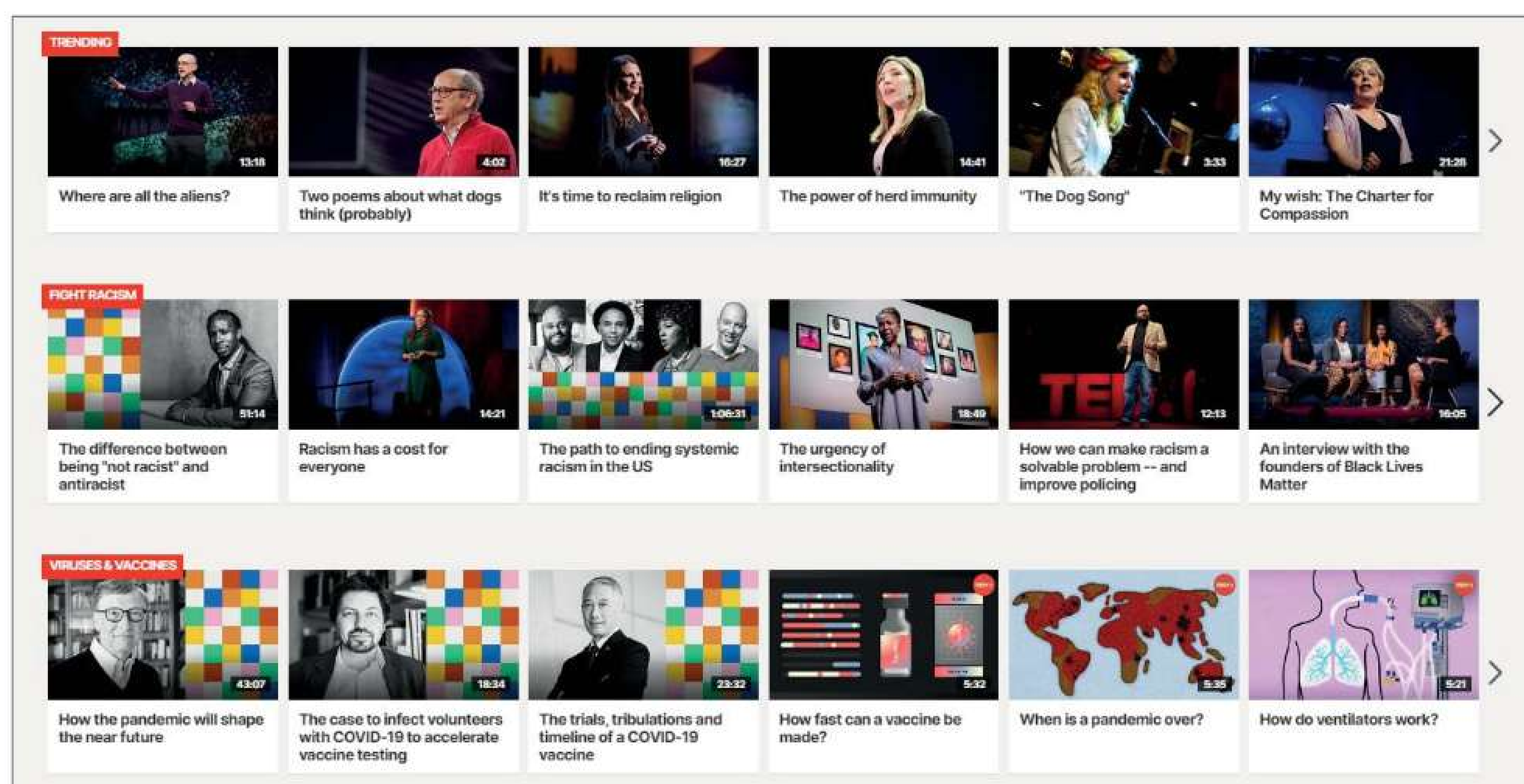
Wyniki badań internautów są publikowane na stronie **globeatnight.org**. Tam też można znaleźć bardziej szczegółowe instrukcje dotyczące prowadzenia obserwacji i korzystania z aplikacji na smartfony, która ułatwia zbieranie i przesyłanie danych.

Rys. 6. Fragment formularza obserwacji

TED.COM

TED to akronim nazwy cyklu konferencji *Technology, Entertainment, Design* (czytaj: teknolodżi, entetejnment, dizajn; technologia, rozrywka, projektowanie), które obecnie odbywają się w wielu krajach na całym świecie. Ideą przewodnią przedsięwzięcia jest hasło *Ideas worth spreading* (czytaj: ajdijez łerf spredyn; pomysły godne rozpowszechnienia).

Formuła konferencji polega na krótkich (6–18 minut) wystąpieniach interesujących ludzi, którzy prezentują publiczności ciekawe pomysły. Prelegentami są naukowcy, politycy, biznesmeni, artyści, a nawet magicy.



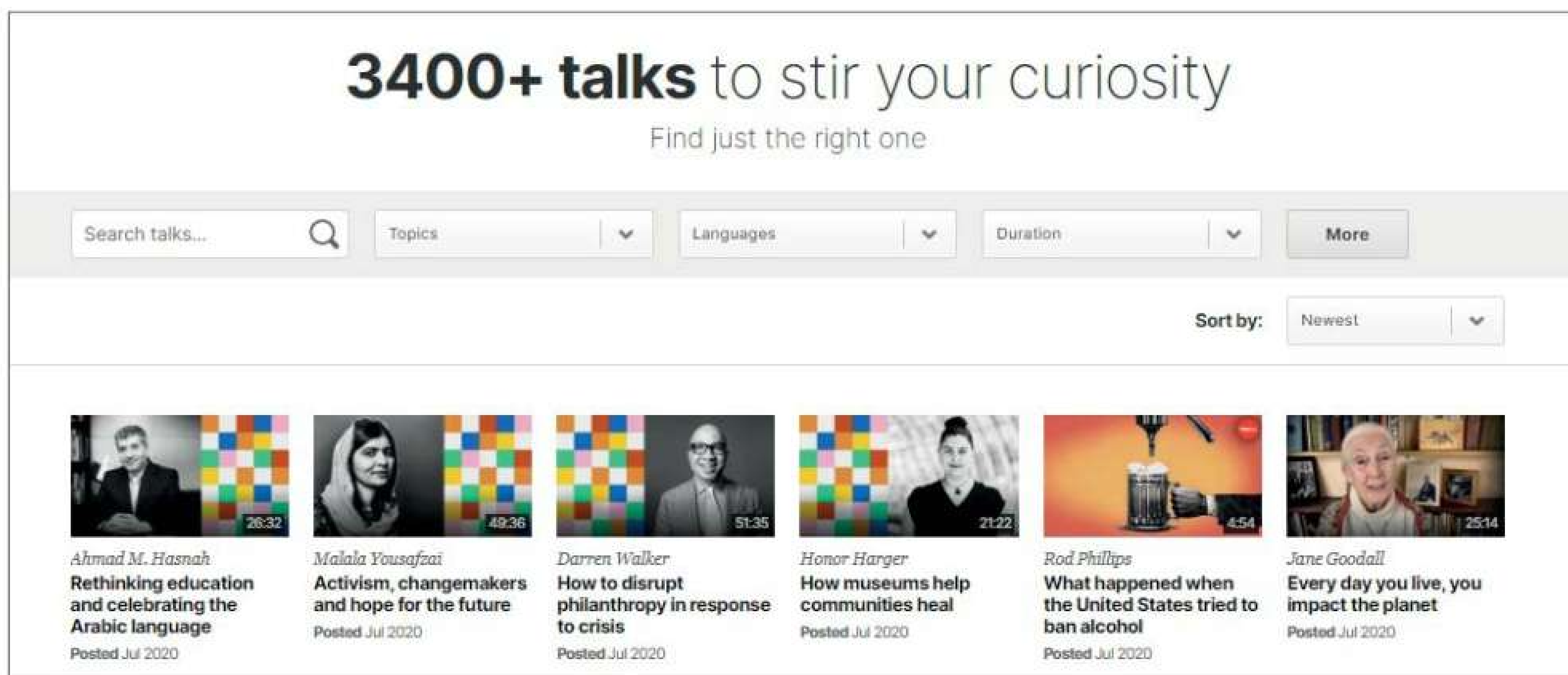
Rys. 7. Fragment strony głównej serwisu TED

Filmy z prezentacjami są publikowane w serwisie TED oraz na profilu TED w serwisie YouTube. Oto pięć spośród ponad 3400 wystąpień, które warto obejrzeć.

- Bill Gates: *How the pandemic will shape the near future* (czytaj: hał di pandemik łil szejp di nijer fjuczer; jak pandemia ukształtuje najbliższą przyszłość).
- Arthur Benjamin: *A performance of „Mathemagic"* (czytaj: e performens of mafemadżik; przedstawienie „matemagii") – matematyk konkuruje z kalkulatorami w mnożeniu dużych liczb i pokazuje, w jaki sposób robi obliczenia.
- Steve Jobs: *How to live before you die* (czytaj: hał tu liw bifor ju daj; jak żyć, zanim umrzesz) – współtwórca firmy Apple opowiada o najważniejszych momentach w swoim życiu.
- Bobby McFerrin: *Watch me play... the audience!* (czytaj: łocz mi plej di odiens; zobaczcie, jak gram... na publiczności) – znany muzyk prezentuje publiczności skalę pentatoniczną (zbudowaną z pięciu dźwięków), co prowadzi do zadziwiających rezultatów.
- Papież Franciszek: *Why the only future worth building includes everyone* (czytaj: łaj di ołnli fjuczer łerf bilding inkludz ewryłan; dlaczego jedyna przyszłość warta tworzenia

obejmuje nas wszystkich) – pierwsze w historii wystąpienie papieża na konferencji TED. Piękne przesłanie dla ludzi z całego świata.

Serwis TED umożliwia wyszukiwanie tematów na różne sposoby. Możesz wybrać filmy najczęściej oglądane lub traktujące o interesujących cię zagadnieniach. Zazwyczaj filmy mają polskie napisy. Aplikacja TED umożliwia pobranie filmów na smartfon lub tablet i oglądanie ich w dowolnym momencie w trybie offline.



Rys. 8. Wyszukiwanie filmu w serwisie TED

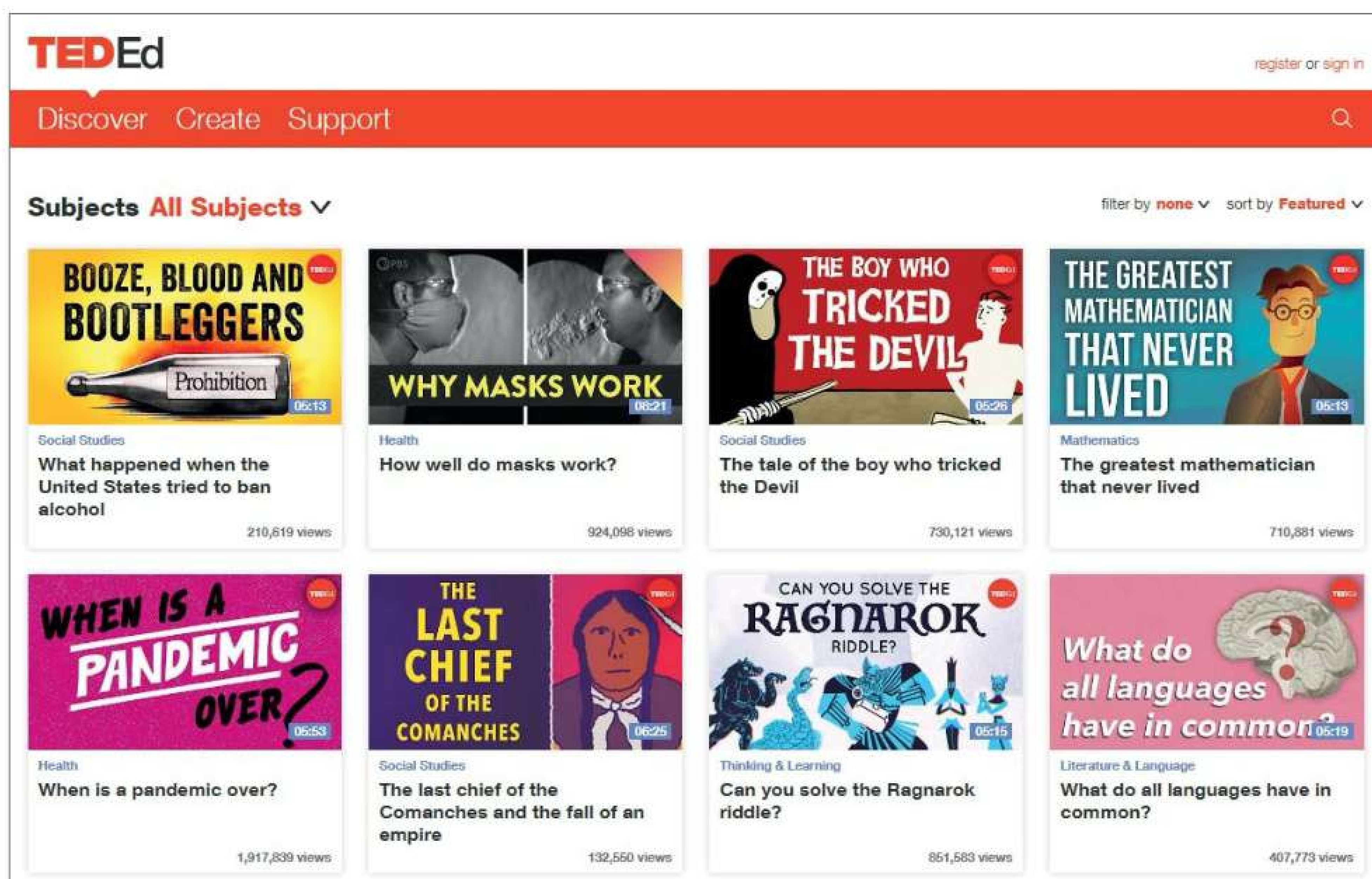
ED.TED.COM

Amerykańscy nauczyciele i naukowcy, zafascynowani wystąpieniami na konferencjach TED, utworzyli portal TEDEd (**ed.ted.com**). Jego hasło to *Lessons worth sharing* (czytaj: lesynz łerf szeryn; lekcje warte rozpowszechniania).

Członkowie zespołu TEDEd wyszukują autorów, w tym nauczycieli, i pomagają im w tworzeniu krótkich filmów-lekcji. Portal często wykorzystuje również fragmenty wystąpień z konferencji TED. Prezentacje umieszczane na portalu są bardzo różnorodne, nie wszystkie mają polskie napisy. Szczególnie warto zobaczyć siedem poniższych realizacji.

- Adam Savage: *How Simple ideas lead to scientific discoveries* (czytaj: hał simply ajdijez lid tu sajentyfik diskaweriz; jak proste pomysły prowadzą do odkryć naukowych).
- Chris Anderson: *Questions no one knows the answer to* (czytaj: kłeszczynz nołłan nołz di anser tu; pytania, na które nikt nie zna odpowiedzi).
- Jonathan Bergmann: *Just how small is an atom?* (czytaj: dżast hał smol yz en etem; jak mały jest atom).
- John Lloyd: *What's invisible? More than you think* (czytaj: łots ynwyzybyl? mor den ju fink; co jest niewidzialne? więcej niż myślisz).
- Dennis Wildfogel: *How big is infinity?* (czytaj: hał big yz ynfynyty; jak wielka jest nieskończoność).

- Chris Anderson: *Why can't we see evidence of alien life?* (czytaj: łaj kant łi si ewidens ow alien lajf; dlaczego nie dostrzegamy dowodów istnienia obcej cywilizacji).
- Tierney Thys: *The secret life of plankton* (czytaj: di sikret lajf ow plankton; sekretne życie planktonu).



Rys. 9. Fragment strony z lekcjami serwisu TEDEd

ZADANIA

1. Wybierz jeden z projektów dostępnych w serwisach SciStarter bądź Zooniverse. Pracuj w nim przez miesiąc w wolnym czasie. Przedstaw opracowane wyniki, swój udział i opinię o projekcie na forum klasy.

2. Umieśćcie adresy ciekawych serwisów i stron WWW w dostępnym dla całej klasy dokumencie Google. W każdej chwili będziecie mogli skorzystać z wpisanych tam adresów, dopisać nowe, ciekawe miejsce w sieci i poprawić nieaktualne linki.

5.2 Ucz się informatyki w sieci

DOWIESZ SIĘ,

- czym jest MOOC,
- w jaki sposób można wykorzystać serwis Akademii Khana do nauki informatyki.

Akademię Khana już trochę znasz – z lekcji *Akademia matematyki* w klasie 6, z lekcji *Czy masz 1101 lat, Liczby parzyste, liczby pierwsze, liczby..., Wybieranie, sortowanie* i *Euklides poprawiony* w klasie 7 oraz tegorocznych lekcji poświęconych tworzeniu stron WWW. Teraz zobaczysz, w jaki sposób można w pełni wykorzystać potencjał Akademii i samodzielnie pogłębiać wiedzę informatyczną.

MOOC

Szkolenia, w których można brać udział bez konieczności obecności na sali wykładowej, są oferowane przez różne instytucje edukacyjne, nierzadko w formie masowych otwartych kursów online, czyli MOOC (ang. *Massive Open Online Courses*; czytaj: masiw ołpen onlajn kerses). Otwartość nie zawsze oznacza, że kursy są bezpłatne. Bywa, że ogólnodostępne materiały ograniczają się do klasycznego wykładu w formie filmu oraz automatycznie sprawdzanego testu, a trzeba zapłacić za dodatkowe ćwiczenia, prace kontrolne albo kontakt z żywymi prowadzącymi czy też egzamin umożliwiający otrzymanie dyplomu potwierdzającego ukończenie kursu.

Do największych międzynarodowych firm oferujących MOOC należą Coursera (w katalogu ma ponad 2000 kursów m.in. w językach angielskim, hiszpańskim, rosyjskim i mandaryńskim), Udacity (głównie kursy dokształcające dla profesjonalistów) i edX (ponad 1200 kursów na poziomie uniwersyteckim). Polskie platformy udostępniające tego typu szkolenia to np. Copernicus College, Navoica.

▶ Wejdź na stronę biblioteki Politechniki Gdańskiej (**pg.edu.pl/biblioteka-pg/mooc**) i zapoznaj się z zaletami i wadami kursów e-learningowych. Wypisz kilka najważniejszych dla ciebie zalet takich kursów.

E-LEARNING W AKADEMII KHANA

Bezpłatne kursy MOOC dla młodzieży oferuje Akademia Khana (**pl.khanacademy.org**), która powoli staje się największą uczelnią na świecie – dostępną dla uczniów w różnym wieku, wszędzie tam, gdzie dociera internet. Choć najwięcej zasobów dotyczy matematyki, coraz więcej filmów i ćwiczeń poświęca się różnym zagadnieniom z zakresu

przyrody (biologia, chemia, medycyna, fizyka, astronomia), informatyki, ekonomii, nauk humanistycznych (np. muzyka, historia świata, sztuka średniowiecznej Europy).

Przedmioty ▲ Szukaj Khan Academy witek.kranas 9+

MATEMATYKA WG DZIAŁÓW
Początek matematyki 11%
Istota arytmetyki 25%
Wstęp do algebry 19%
Podstawy algebry 19%
Podstawowa geometria 10%
Geometria na poziomie liceum 8%
Trygonometria
Statystyka 3%
Wstęp do rachunku różniczkowego 1%
Rachunek pochodnych i całkowy
Rachunek całkowy
Rachunek różniczkowy wielu zmiennych
Równania różniczkowe

MATEMATYKA WG KLAS
1 klasa (USA) 12%
2 klasa (USA) 3%
3 klasa (USA) 14%
4 klasa (USA) 11%
5 klasa (USA) 4%
6 klasa (USA) 12%
7 klasa (USA) 8%
8 klasa (USA) 5%

NAUKI ŚCISŁE
Biologia
Fizyka
Chemia 6%
Kosmologia i astronomia
Zdrowie i medycyna
Chemia organiczna

EKONOMIA I FINANSE
Mikroekonomia
Makroekonomia
Finanse i rynki kapitałowe
Przedsiębiorczość

NAUKI HUMANISTYCZNE I SZTUKA
Historia sztuki
Historia sztuki - kurs rozszerzony
Gramatyka angielska
Muzyka
Storytelling, czyli sztuka opowiadania

HISTORIA ŚWIATA
Historia świata
AP® Historia Świata
Historia USA
AP® US Historia
Rząd Stanów Zjednoczonych i obywatele

INFORMATYKA
Programowanie
Informatyka
Godzina kodowania
Pixar w pigułce

PRZYGOTOWANIE DO EGZAMINU
SAT (po angielsku)
LSAT (po angielsku)

Rys. 1. Strona **Przedmioty** w serwisie Akademii Khana

Informatyka

Godzina kodowania
Dodatkowe źródła

Programowanie
Wprowadzenie do JS: Rysowanie i animacja
Wstęp do HTML/CSS: Tworzenie stron int...
Wstęp do SQL: Zarządzanie danymi i twor...
Zaawansowany JS: Gry i Wizualizacje
Zaawansowany JS: Symulacja natury
HTML/JS: Tworzenie interaktywnych stron...
HTML/JS: Tworzenie interaktywnych stron...
Poznaj profesjonalistów

Informatyka
Algorytmy
Podróż w krainę kryptografii
Podróż do teorii informacji

Podstawy informatyki - program rozszerzony
Cyfrowe zapisywanie informacji
Internet
Programowanie
Algorytmy
Analiza danych
Przygotowanie do egzaminu (chodzi o SAT,...
Odwzorowanie treści tego rozdziału na sta...

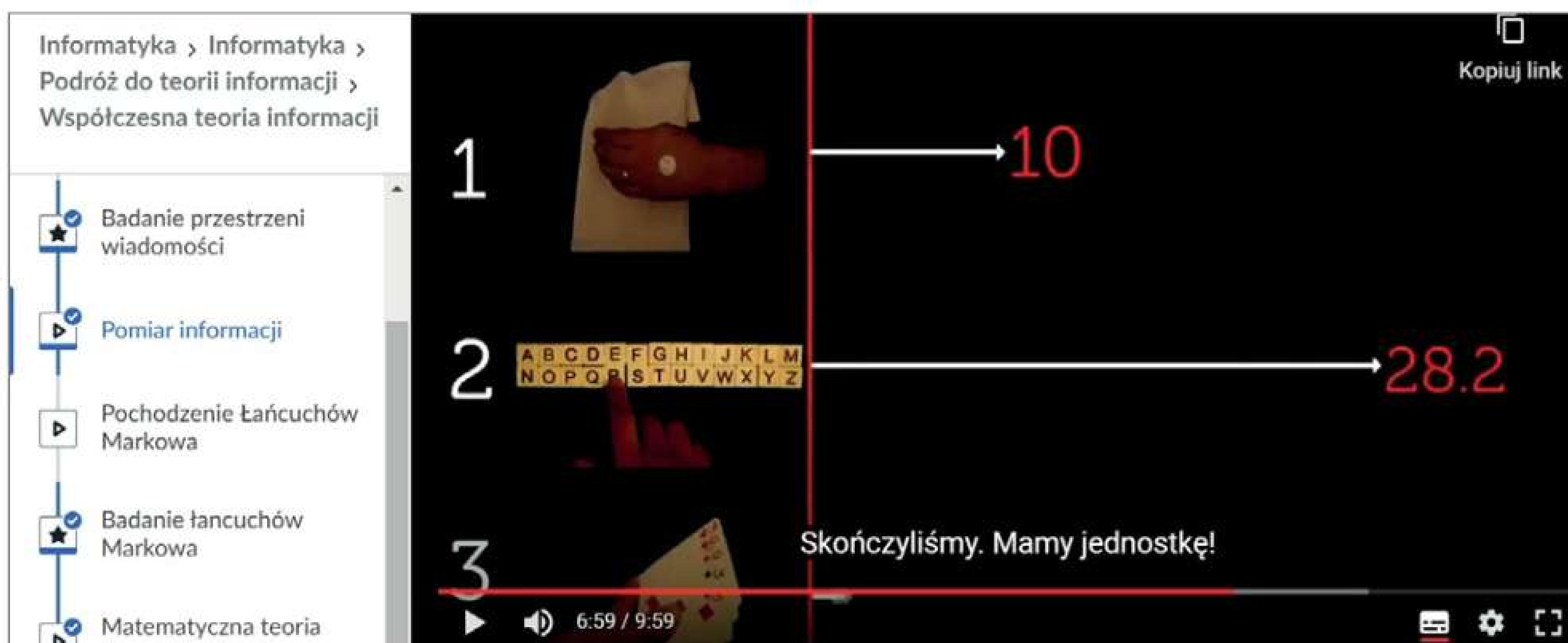
Rys. 2. Lista kursów przedmiotu Informatyka

W ramach przedmiotu **Informatyka** wyróżniono dwa działy – *Programowanie* i *Informatyka* – oraz kursy *Godzina kodowania* i *Podstawy informatyki – program rozszerzony*. Na głównej stronie przedmiotów można też znaleźć kurs *Pixar w pigułce*, realizowany we współpracy z wytwórnią filmową Pixar Animation Studios (czytaj: piksar animejszyn studios).

W dziale **Informatyka** zamieszczono trzy kursy. Dwa pierwsze – *Algorytmy* i *Podróż w krainę kryptografii* – są przeznaczone dla zaawansowanych użytkowników, którzy potrafią już programować w języku JavaScript. Oba są interesujące i dostępne dla każdego, kogo zaciekawi tematyka.

Trzeci kurs, *Podróż do teorii informacji*, omawia teorię informacji od czasów starożytnych do opracowania kodu Morse'a, a także zagadnienia związane ze współczesną teorią informacji, np. pomiar informacji.

► Wybierz kurs *Podróż do teorii informacji* i obejrzyj film wprowadzający w temat *Pomiar informacji*.



Rys. 3. Zrzut strony z filmem *Pomiar informacji*

Dział **Programowanie** rozwija się bardzo szybko i wykorzystuje specjalne środowisko umożliwiające interaktywną pracę z kodem. Obecnie jest dostępnych siedem kursów z tego zakresu. Jeden już trochę znasz – to *Wstęp do HTML/CSS: Tworzenie stron internetowych* z lekcji 1.1–1.5. Pozostałe szkolenia dotyczą języka JavaScript (JS), tworzenia stron internetowych w HTML/CSS oraz języka SQL:

- *Wprowadzenie do JS: Rysowanie i animacja*,
- *Zaawansowany JS: Gry i Wizualizacje*,
- *Zaawansowany JS: Symulacja natury*,
- *Wstęp do HTML/JS: Tworzenie interaktywnych stron internetowych*,
- *HTML/JS: Tworzenie interaktywnych stron internetowych z jQuery*,
- *Wstęp do SQL: Zarządzanie danymi i tworzenie zapytań*.

Interesujący dodatek stanowi kurs *Poznaj profesjonalistów*, który pokazuje, w jaki sposób można wykorzystać zdobyte umiejętności informatyczne i programistyczne.

Może zaciekawi cię kurs *Wprowadzenie do JS: Rysowanie i animacja*? Jest dość obszerny: zaczyna się od wyjaśnienia, na czym polega programowanie, omówienia środowiska, wprowadzenia do rysowania i kolorowania. Kolejne sekcje to m.in.: **Zmienne**, **Podstawy animacji**, **Programy interaktywne**, **Tekst i ciągi znaków**, **Funkcje**, **Logika i instrukcja if**, **Pętle**, **Tablice**, **Obiekty**, a na końcu **Stawanie się coraz lepszym programistą**. Możesz pracować we własnym tempie, w dowolnym czasie.

Nowy dział *Podstawy informatyki – program rozszerzony* jest dostosowany do wymagań amerykańskiego certyfikowanego kursu Computer Science Principals. Ciekawostką jest możliwość porównania zapisu rozwiązania problemu w kilku środowiskach programistycznych.

```
for (var i = 1; i <= 10; i++) {
  for (var j = 1; j <= 10; j++) {
    println(i * j);
  }
}
```

Run code | See similar code in: App Lab | Snap | Python

JavaScript, środowisko Akademii Khana

@PamelaFox1/NumberedRepetitionMultiples
No description
Files
main.py

```
for i in range(1, 11):
  for j in range(1, 11):
    print(i * j)
```

Python, środowisko repl.it



Bloki w środowisku Snap

Zobacz tylko:

```
for (var i = 1; i <= 10; i++) {
  for (var j = 1; j <= 10; j++) {
    console.log(i * j);
  }
}
```

Bloki w środowisku App Lab

Rys. 4. Realizacja tabliczki mnożenia za pomocą zagnieżdżonych pętli

NAUKA W AKADEMII KHANA – SAMODZIELNA I Z TRENEREM

W Akademii Khana można uczyć się samodzielnie albo w sposób zorganizowany (tematy do zrealizowania wyznacza trener – rodzic, opiekun lub nauczyciel). Zobacz, jak pracuje się w klasie.

► Zarejestruj się w Akademii Khana lub zaloguj się, jeśli rejestrację masz już za sobą.

► Zapisz się do klasy w Akademii Khana. Na stronie ucznia kliknij zakładkę **Nauczyciele** i w oknie **Dołącz do klasy** wpisz kod klasy, a następnie kliknij przycisk **Dołącz do klasy**. Jeśli nauczyciel nie podał ci kodu, możesz wpisać kod klasy utworzonej przez autorów podręcznika: **82FRHG**.

Rys. 5. Zapisywanie się do klasy w serwisie Akademii Khana

► Sprawdź, jakie tematy wybrał dla ciebie nauczyciel. Kliknij nazwę użytkownika w prawym górnym rogu i wybierz powiadomienia, a potem kliknij wybrany kurs.

Trener otrzymuje sprawozdanie dotyczące twojej aktywności – widzi systematyczność i tempo pracy, liczbę podejść do zadań. W ten sposób łatwiej mu dostrzec twoje mocne i słabe strony, określić przyczyny ewentualnych niepowodzeń, a w razie potrzeby zindywidualizować tok nauczania (trener może zadawać filmy do obejrzenia, artykuły do przeczytania czy ćwiczenia do rozwiązania zarówno całym klasom, jak i pojedynczym uczniom). W zakładce **Postępy** możesz sprawdzić swoje odznaki i statystyki.

ZADANIA

1. Poszukaj szkoleń prowadzonych w formie MOOC w języku polskim. Porównaj wybranego dostawcę polskiego i międzynarodowego.

2. Po obejrzeniu filmu *Pomiar informacji* przypomnij sobie system dwójkowy omawiany w klasie 7 (lekcja 1.2. *Czy masz 1101 lat*). Zamień na postać dwójkową liczbę liter w alfabecie angielskim (26), liczbę liter w alfabecie polskim (35) i liczbę kart w talii (52).

5.3 Tak daleko, tak blisko

DOWIESZ SIĘ, JAK

- założyć konto na stronie programu Mikogo,
- rozpocząć sesję i zaprosić do współpracy innych użytkowników programu Mikogo,
- dołączyć do istniejącej sesji,
- udostępniać pulpit uczestnikom sesji, np. w celu pracy nad wspólnym projektem.

Co zrobić, kiedy w czasie pracy z komputerem zaistnieje konieczność skorzystania z czyjejś pomocy, a w pobliżu nie ma nikogo, kto mógłby udzielić wsparcia? Z jakiego programu skorzystać podczas zdalnej pracy nad wspólnym projektem? O tym właśnie jest ta lekcja.

GDZIE SZUKAĆ POMOCY

Książki, kursy i poradniki nie zawsze dostarczają szukanych informacji – ich autorzy nie są w stanie przewidzieć wszystkich problemów, z którymi możesz się spotkać. Warto wtedy poszukać pomocy w internecie, zajrzeć na portal społecznościowy albo tematyczne forum dyskusyjne – np. użytkownicy serwisu **www.forumweb.pl** oferują porady dotyczące m.in. sprzętu, oprogramowania, systemów mobilnych czy tworzenia stron WWW. Ale czasem nawet i to nie wystarcza. Zdarza się, że musimy skorzystać ze zdalnej pomocy przez przekazanie osobie zaufanej kontroli nad komputerem.

Taką możliwość oferuje bezpłatny program Mikogo, który umożliwia udostępnianie widoku pulpitu komputera użytkownika na żywo, a także uzyskanie dostępu do systemu operacyjnego drugiej osoby.

ZAKŁADANIE KONTA I POBIERANIE PROGRAMU

Aby używać programu Mikogo, musisz spełnić dwa warunki: po pierwsze musisz założyć konto w serwisie **www.mikogo.com**, po drugie musisz pobrać program i zainstalować go na swoim komputerze.

► Załóż konto – wejdź na stronę **www.mikogo.com**, wypełnij formularz i kliknij przycisk **Let's go!**.

Rys. 1. Zakładanie konta Mikogo

Po założeniu konta pobierz program na swój komputer.

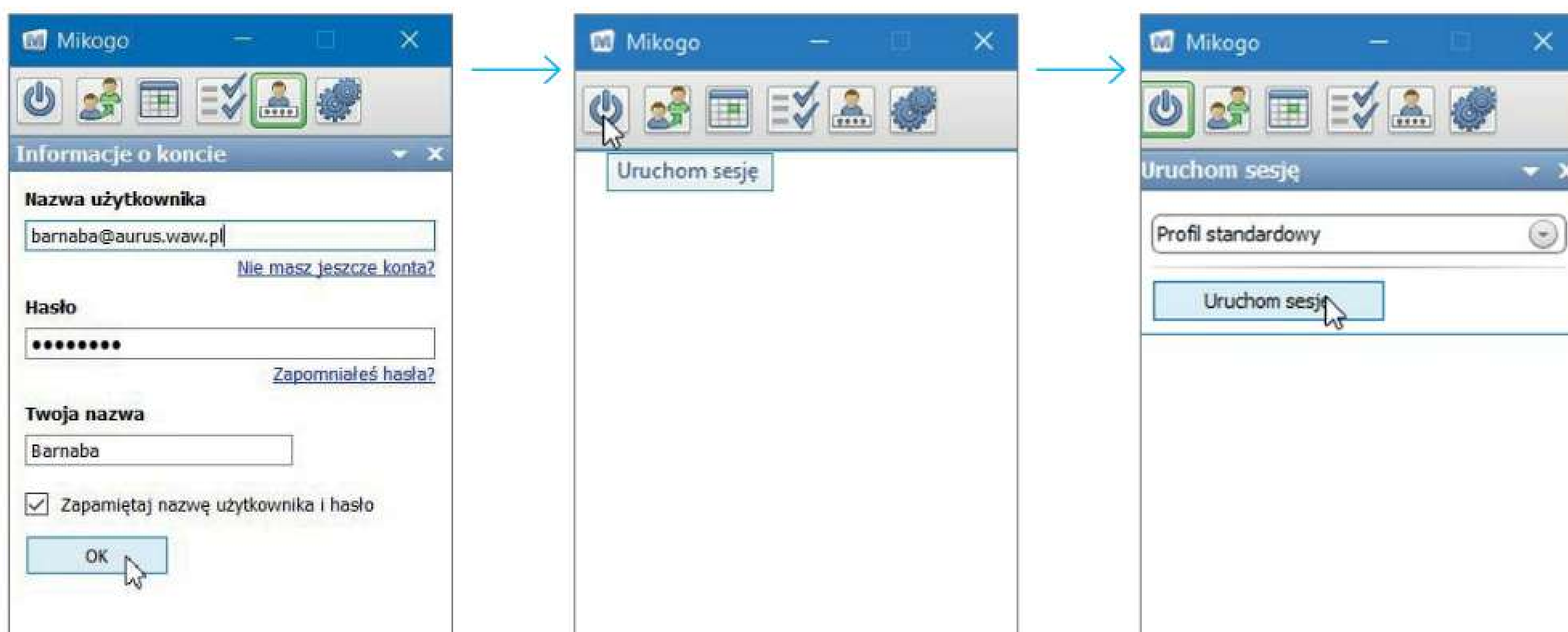
- Kliknij przycisk **DOWNLOAD FOR WINDOWS**.
- Pobrany program zainstaluj na komputerze. (Do tej operacji potrzebne są uprawnienia administratora. Jeśli nie masz takich uprawnień, nauczyciel powinien ci ich udzielić).

Warto zaznaczyć, że ze strony **www.mikogo.com** można pobrać wersję programu Mikogo do pracy w różnych systemach operacyjnych, jak również na urządzenia mobilne pracujące pod kontrolą systemu Android oraz Ios. Poza wersją przeznaczoną do systemu Windows dostępna jest też wersja do MacOs.

W czasie tej lekcji zostanie omówiona wersja podstawowa (darmowa) programu, ale jest też wersja rozszerzona (płatna). W wersji płatnej można korzystać z tak zwanej opcji premium, dającej użytkownikowi więcej możliwości.

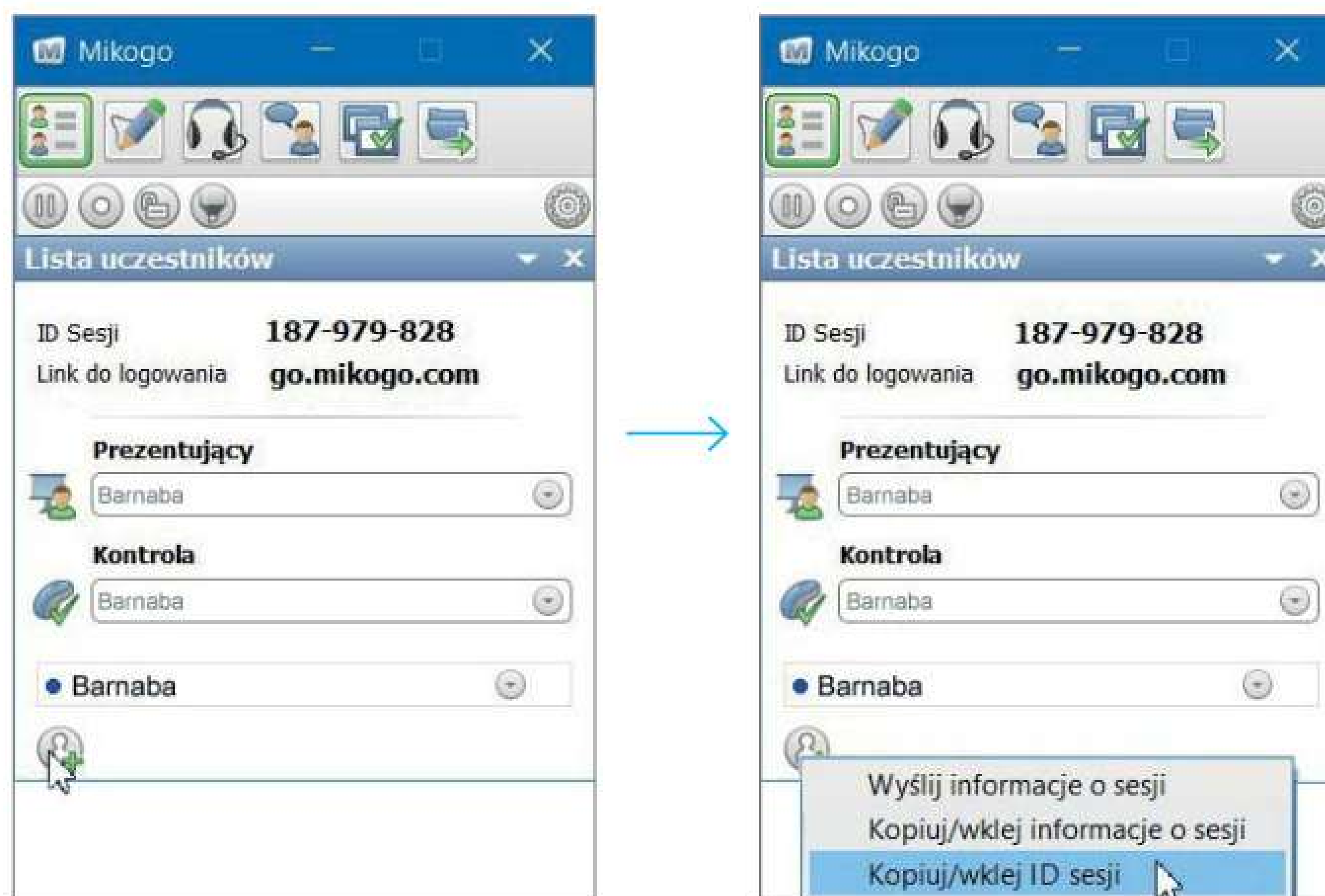
URUCHOMIENIE SESJI I ZAPROSZENIE UCZESTNIKÓW

Aby rozpocząć sesję, trzeba uruchomić program i zalogować się na swoje konto, potem wystarczy kliknąć kolejno ikonę **Uruchom sesję** na pasku narzędzi i przycisk **Uruchom sesję**.



Rys. 2. Uruchamianie sesji

Po uruchomieniu sesji można zaprosić uczestnika lub uczestników sesji. W tym celu należy kliknąć ikonę **Zaproszenie**, a następnie z rozwiniętej listy wybrać pozycję **Kopiuj/wklej ID sesji**. Numer ID sesji zostanie skopiowany do schowka. Można go wysłać e-mailem, esemesem albo po prostu przekazać w rozmowie telefonicznej.



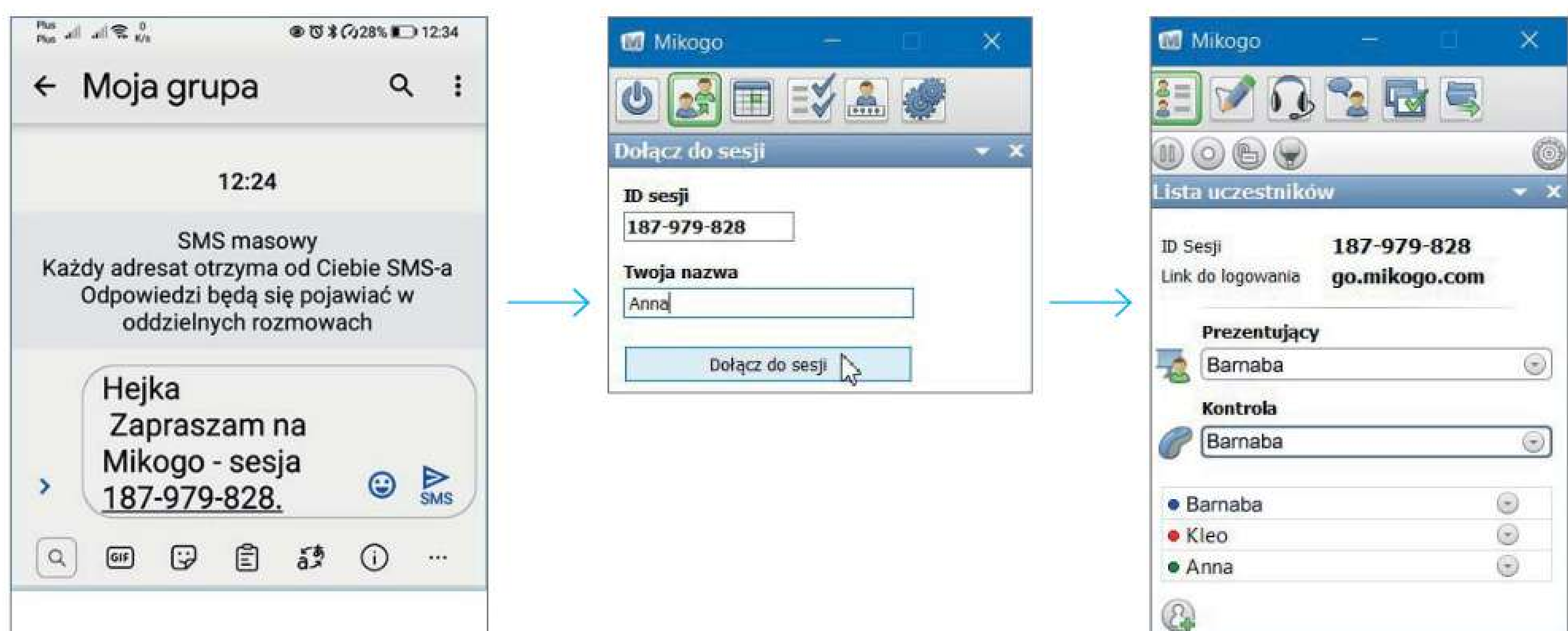
Rys. 3. Zapraszanie uczestników sesji

Prezentującym sesji 187-979-828, czyli uczestnikiem sesji strumieniującym obraz pulpitu swojego komputera, jest Barnaba. Ponieważ to on rozpoczął sesję, on decyduje o tym, kto sprawuje kontrolę nad udostępnianym komputerem – w tym przypadku kontrolę przyznał sobie. Jeśli zechce, będzie mógł przekazać kontrolę dowolnemu uczestnikowi spotkania.

DOŁĄCZENIE DO SESJI

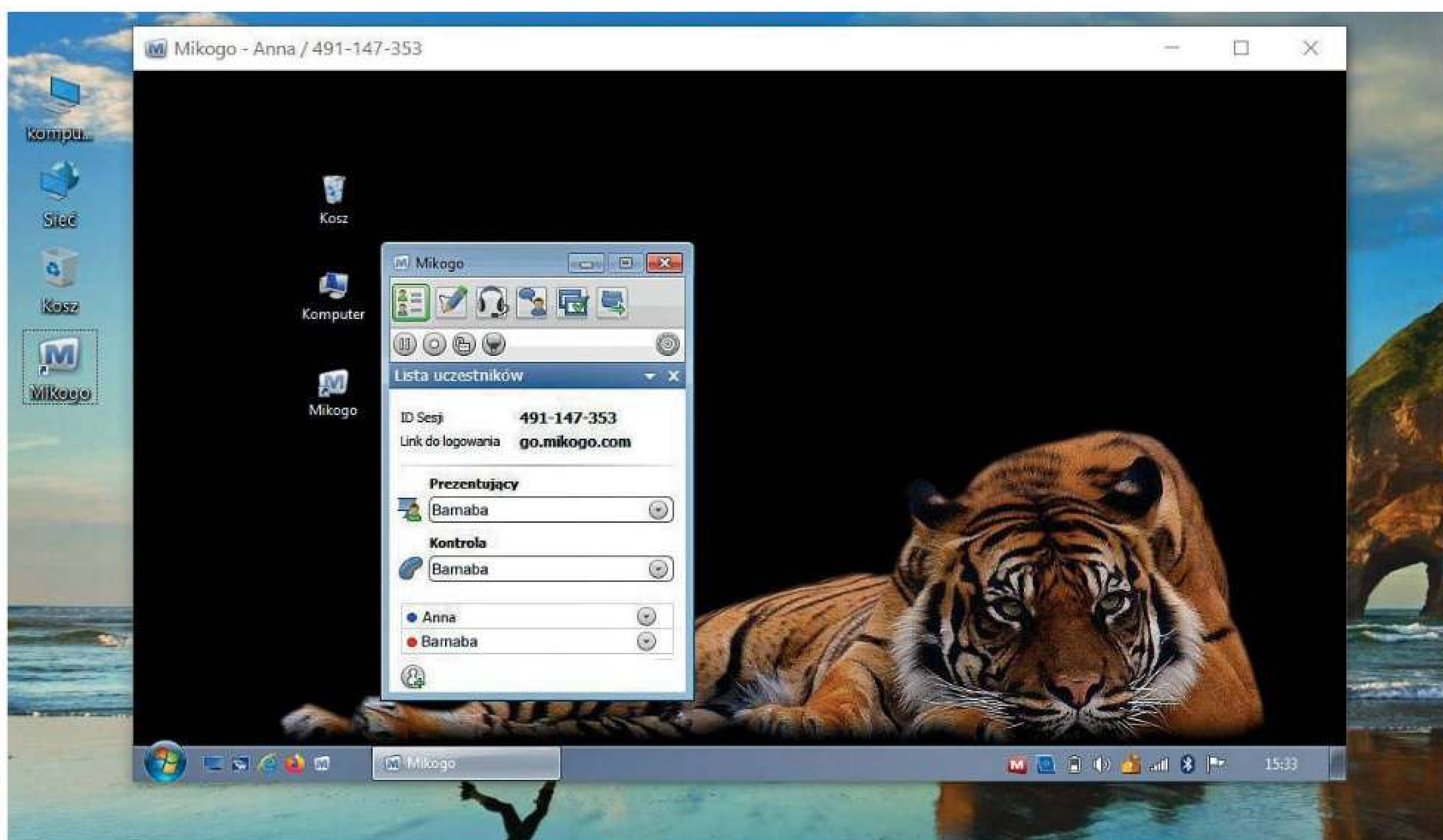
Osoba, która otrzymała zaproszenie do udziału w sesji i chce w niej uczestniczyć, powinna wykonać następujące czynności:

- uruchomić program Mikogo,
- wpisać otrzymany numer ID sesji,
- wpisać swoją nazwę,
- kliknąć przycisk **Dołącz do sesji**.



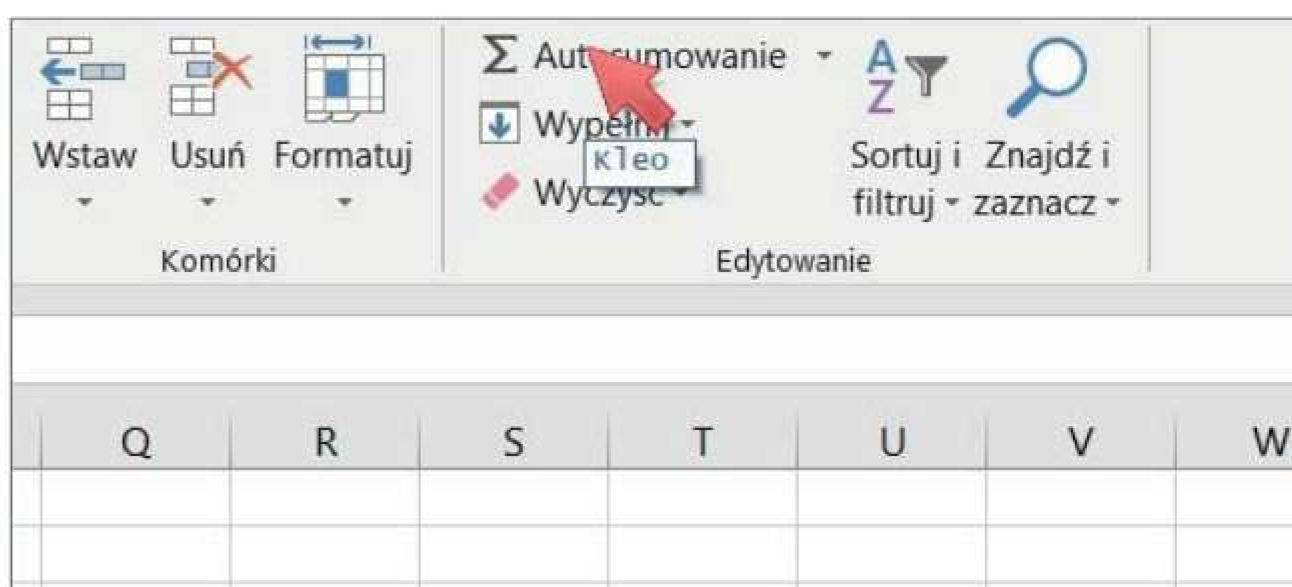
Rys. 4. Dołączanie do sesji – widok esemesa wysłanego przez Barnabę do grupy odbiorców oraz okna Mikogo osoby zapraszanej (Anna). Oprócz Anny do sesji dołączył również Kleo

SESJA
Podczas sesji uczestnicy spotkania widzą pulpit osoby prowadzącej spotkanie.



Rys. 5. Tak pulpit Barnaby widzi Anna na swoim komputerze

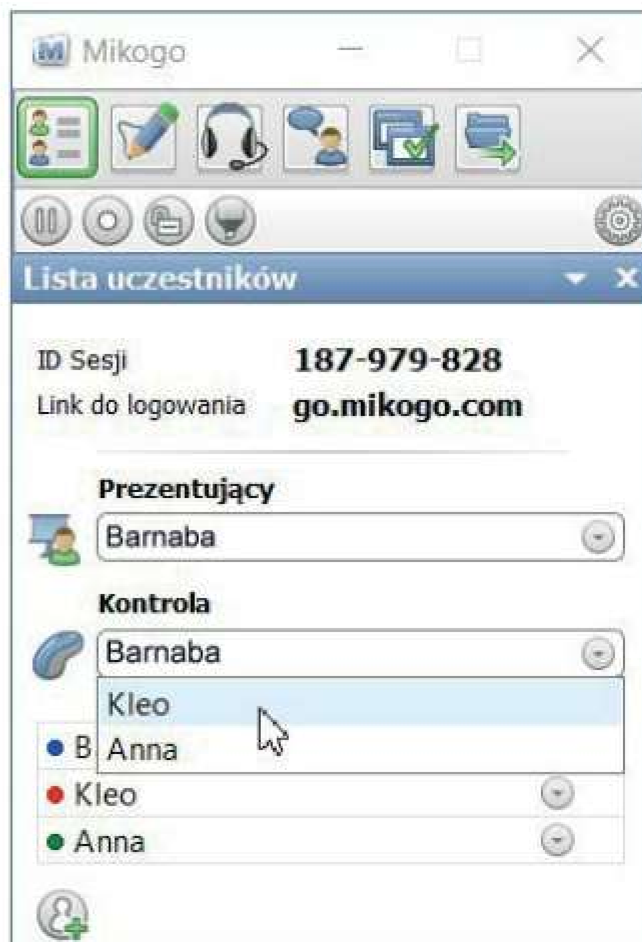
Gdy uczestnik sesji sprawuje kontrolę nad swoim pulpitem, pozostali mogą jedynie wskazywać obiekty – klikać je lewym przyciskiem myszy.



Rys. 6. Kleo pokazuje Barnabie, jakiego narzędzia powinien użyć, aby zsumować zawartość zaznaczonych komórek w arkuszu kalkulacyjnym

Jeśli prezentujący połączył się z uczestnikiem, ponieważ nie radzi sobie z wykonaniem jakichś czynności na komputerze, np. nie potrafi zmienić jasności ekranu, może przekazać temu uczestnikowi kontrolę nad swoim komputerem. Wówczas uczestnik może wykonać te czynności na komputerze prezentującego.

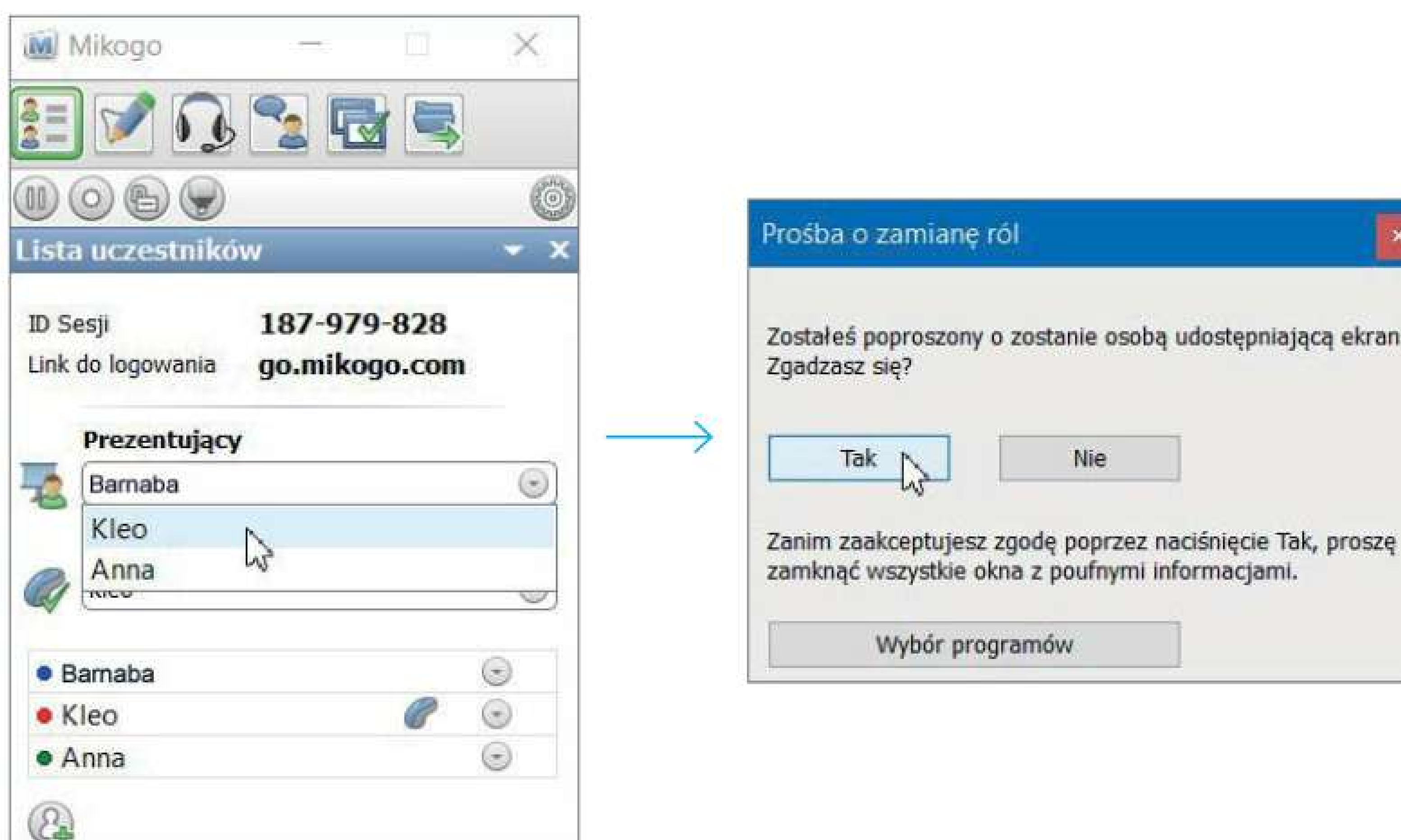
Aby przekazać kontrolę, należy rozwinąć listę **Kontrola** i wskazać osobę, której przekazuje się kontrolę nad komputerem.



Rys. 7. Barnaba przekazuje kontrolę nad swoim komputerem uczestnikowi sesji Kleo

Po przejęciu kontroli nad komputerem uczestnik sesji może pracować na nim tak jak na swoim własnym. Właściciel komputera w każdej chwili może odebrać przyznane uprawnienia. W celu odebrania kontroli nad twoim komputerem innemu uczestnikowi sesji musisz nacisnąć kombinację klawiszy **Ctrl+F12**.

Podczas sesji uczestnicy mogą też zmieniać role. Uczestnik prezentujący swój pulpit może poprosić o zmianę osoby prezentującej. Ta operacja jest oczywiście możliwa pod warunkiem, że wywołana osoba wyrazi zgodę na udostępnienie innym uczestnikom sesji swojego komputera.



Rys. 8. Prośba o zmianę komputera udostępniającego swoje zasoby i o zgodę na udostępnienie widoku pulpitu komputera

Naciśnięcie przycisku **Tak** spowoduje, że wszyscy użytkownicy spotkania zobaczą na swoich komputerach obraz pulpitu osoby, która udzieliła zgody. Jednocześnie uczestnik prezentujący przejmuje kontrolę.

To oznacza, że w czasie sesji jej uczestnicy, przekazujący sobie uprawnienia i udostępniający komputery, mogą nie tylko udzielać sobie wzajemnie pomocy, lecz także pracować nad wspólnym projektem.

W czasie sesji jej uczestnicy mogą komunikować się ze sobą głosowo, mogą też korzystać z dostępnego w programie czatu.

ZADANIE

1. Dobierzcie się w grupy trzy- lub czteroosobowe i ustalcie, kto będzie liderem koordynującym działania zespołu. Wykonajcie wspólnie prezentację na temat komunikacji zdalnej w sieci. Porozumiewajcie się między sobą wyłącznie za pośrednictwem programu Mikogo.

5.4 Ze smartfonem na piechotę

DOWIESZ SIĘ, JAK

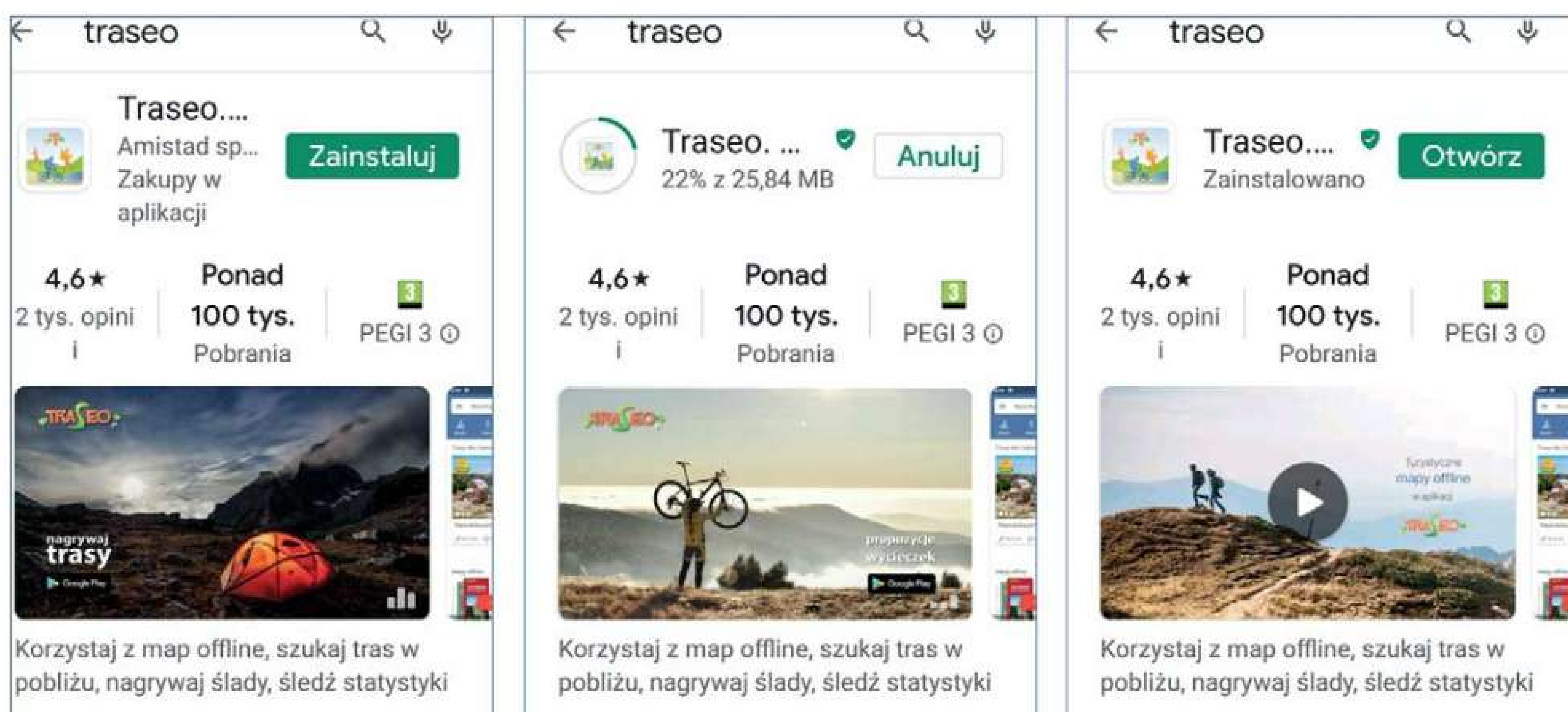
- wykorzystać technologię telekomunikacyjną na wycieczce,
- udokumentować przebieg wycieczki w smartfonie,
- opublikować trasę wycieczki w internecie.

Nowoczesne technologie są powszechnie wykorzystywane w różnych dyscyplinach sportu i wśród osób lubiących aktywne spędzanie czasu. Jedną z aplikacji przeznaczonych dla miłośników mniejszych i większych wypraw jest aplikacja mobilna Traseo, która może być wykorzystywana przez amatorów turystyki pieszej, biegaczy, rowerzystów, kajakarzy czy narciarzy.

SERWIS I APLIKACJA TRASEO

Bezpłatny program Traseo. Szlaki, trasy, mapy pozwala wybrać jedną z kilkudziesięciu tysięcy tras zapisanych przez użytkowników albo zarejestrować własną pieszą wycieczkę, a następnie udostępnić mapę z trasą i opisem na stronie **www.traseo.pl** lub na własnej stronie WWW. Aplikację można pobrać ze Sklepu Play lub z serwisu usługodawcy.

► Pobierz Traseo na swoje urządzenie mobilne – odszukaj aplikację w Sklepie Play i naciśnij przycisk **Zainstaluj.** Uruchom aplikację po zakończeniu jej instalacji.



Rys. 1. Instalowanie aplikacji Traseo

Uwaga: tę aplikację instalujesz na potrzeby lekcji, jednak zanim pobierzesz kolejną aplikację na swoje urządzenie, zadaj sobie pytanie, czy naprawdę warto. Pamiętaj, że na instalację musisz uzyskać zgodę rodziców lub opiekunów.

- Wejdź na stronę **www.traseo.pl/regulamin** – zapoznaj się z regulaminem korzystania z usługi Traseo, zwróć uwagę na to, czego oczekuje od ciebie usługodawca i czego ty od niego możesz oczekiwać.
- Wejdź na stronę **www.traseo.pl/polityka-prywatnosci** – sprawdź, w jakim zakresie usługodawca gwarantuje bezpieczeństwo twoich danych.

TWORZENIE KONTA W APLIKACJI TRASEO

Własne konto w aplikacji Traseo umożliwia publikowanie tras wycieczek oraz ich edycję.

- Uruchom aplikację Traseo na swoim urządzeniu mobilnym.
- Kliknij ikonę ☰ w lewym górnym rogu ekranu – pojawi się okno z opcjami ustawień.
- Kliknij pozycję **Zaloguj**.
- Kliknij pozycję **Zarejestruj się**.
- Podaj nazwę użytkownika, hasło (dla bezpieczeństwa nie podawaj hasła, którego używasz do odbierania poczty) i adres e-mail.
- Zaakceptuj regulamin (nie musisz zaznaczać pozycji dotyczącej zgody na otrzymywanie newslettera) i kliknij pozycję **ZAREJESTRUJ**.

Rys. 2. Tworzenie konta w aplikacji Traseo

Jeśli proces tworzenia konta przebiegł poprawnie, pojawi się karta z przyciskiem umożliwiającym rozpoczęcie rejestrowania trasy. Będzie się pojawiać przy każdym uruchomieniu aplikacji.

NAGRYWANIE TRASY

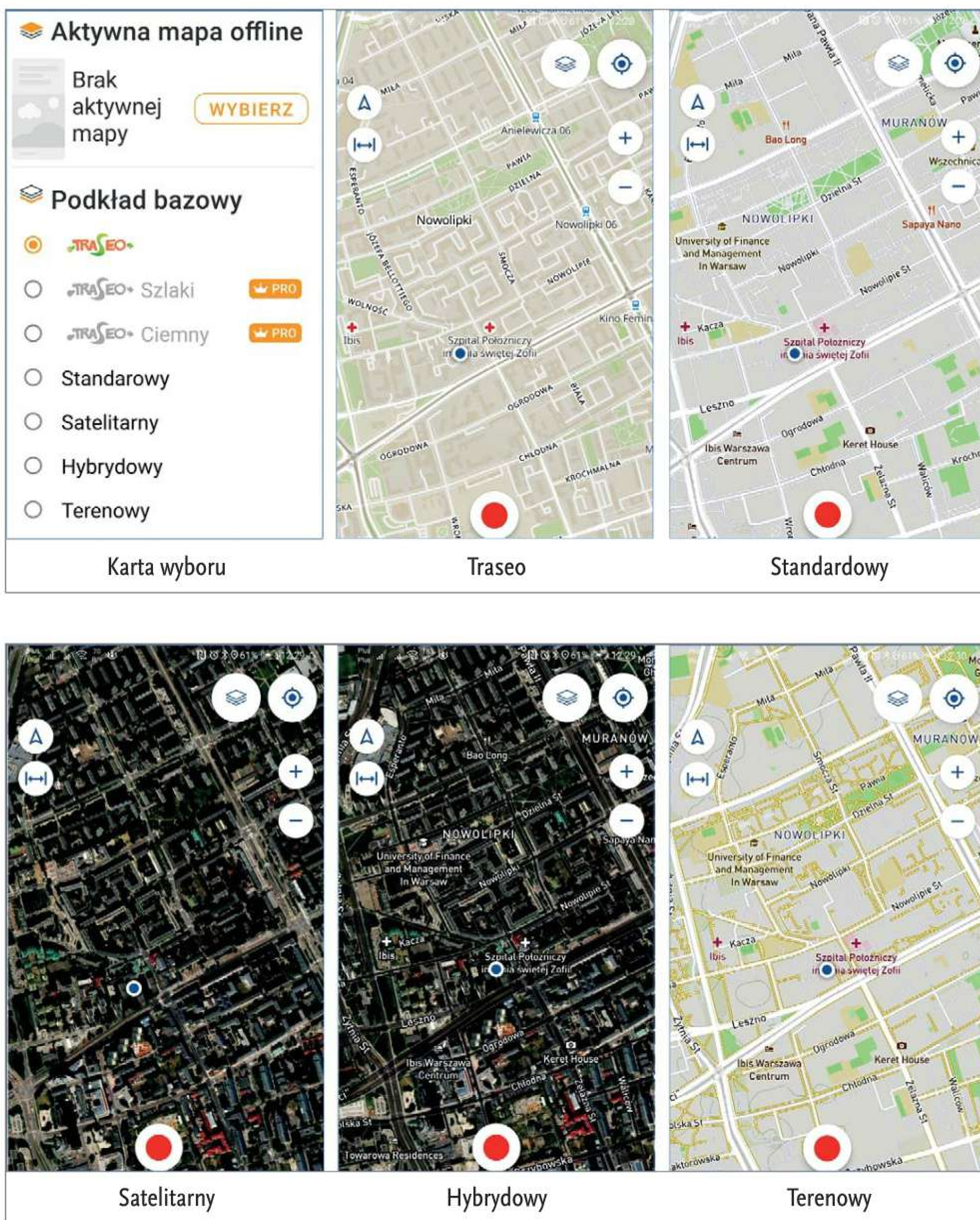
Podczas rejestrowania trasy aplikacja Traseo korzysta z czujnika GPS. Zwiększa to zapotrzebowanie urządzenia mobilnego na energię. Aby uniknąć wyczerpania baterii podczas wycieczki, należy naładować ją wcześniej do pełna i zabrać powerbank – dodatkową baterię podłączaną do urządzenia mobilnego za pośrednictwem kabla USB. Warto również pamiętać, że aplikacja działa tylko online. Oznacza to, że podczas zapisywania trasy urządzenie przez cały czas wysyła i pobiera dane za pośrednictwem internetu. W przypadku określonego limitu ilości przesyłu danych może to spowodować jego przekroczenie i wpłynąć na wysokość rachunku wystawionego przez operatora sieci, w której jest zarejestrowane urządzenie.

Aby rozpocząć nagrywanie trasy, należy dotknąć przycisku **Nagraj trasę**, a następnie wyrazić zgodę na dostęp aplikacji do twojej lokalizacji – czyli dotknąć **ZEZWALAJ TYLKO PODCZAS UŻYWANIA...**.



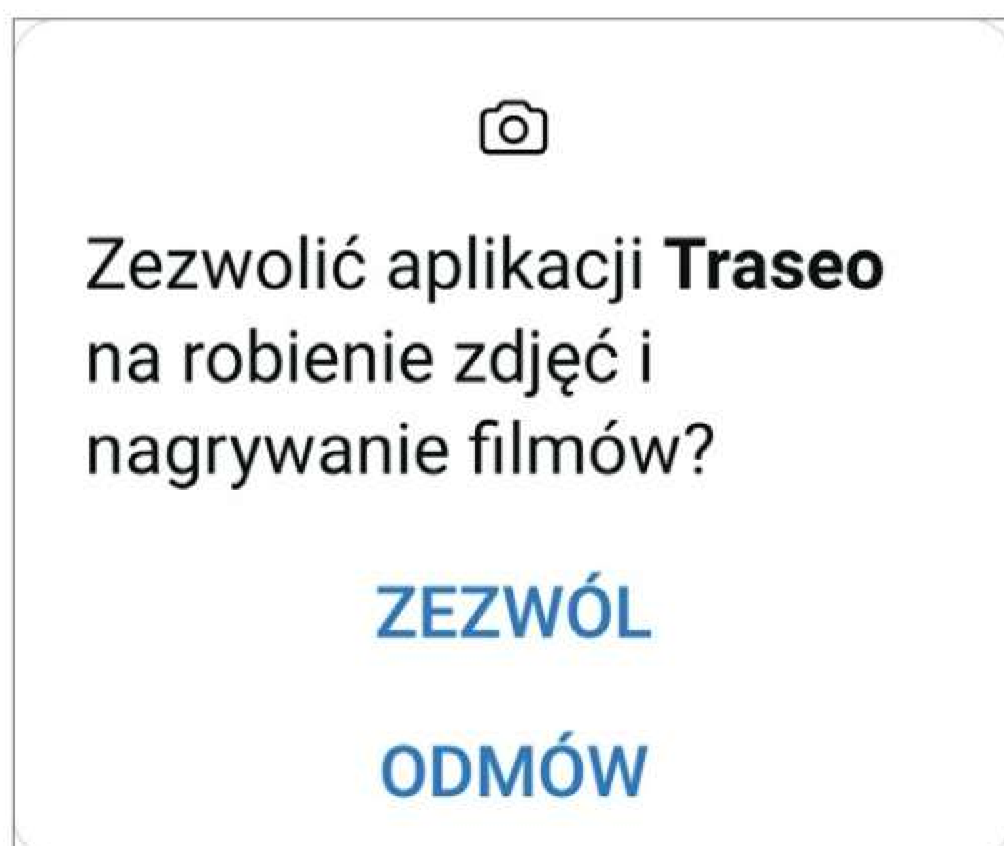
Rys. 3. Pytanie o zezwolenie aplikacji na dostęp do lokalizacji

Gdy wyświetli się mapa z ikoną rozpoczynającą rejestrację trasy i znacznikiem bieżącej lokalizacji, dla większej wygody pracy z aplikacją można wskazać rodzaj mapy (np. **Terenowy**). Menu z wyborem podkładów bazowych wyświetla się po dotknięciu ikony .



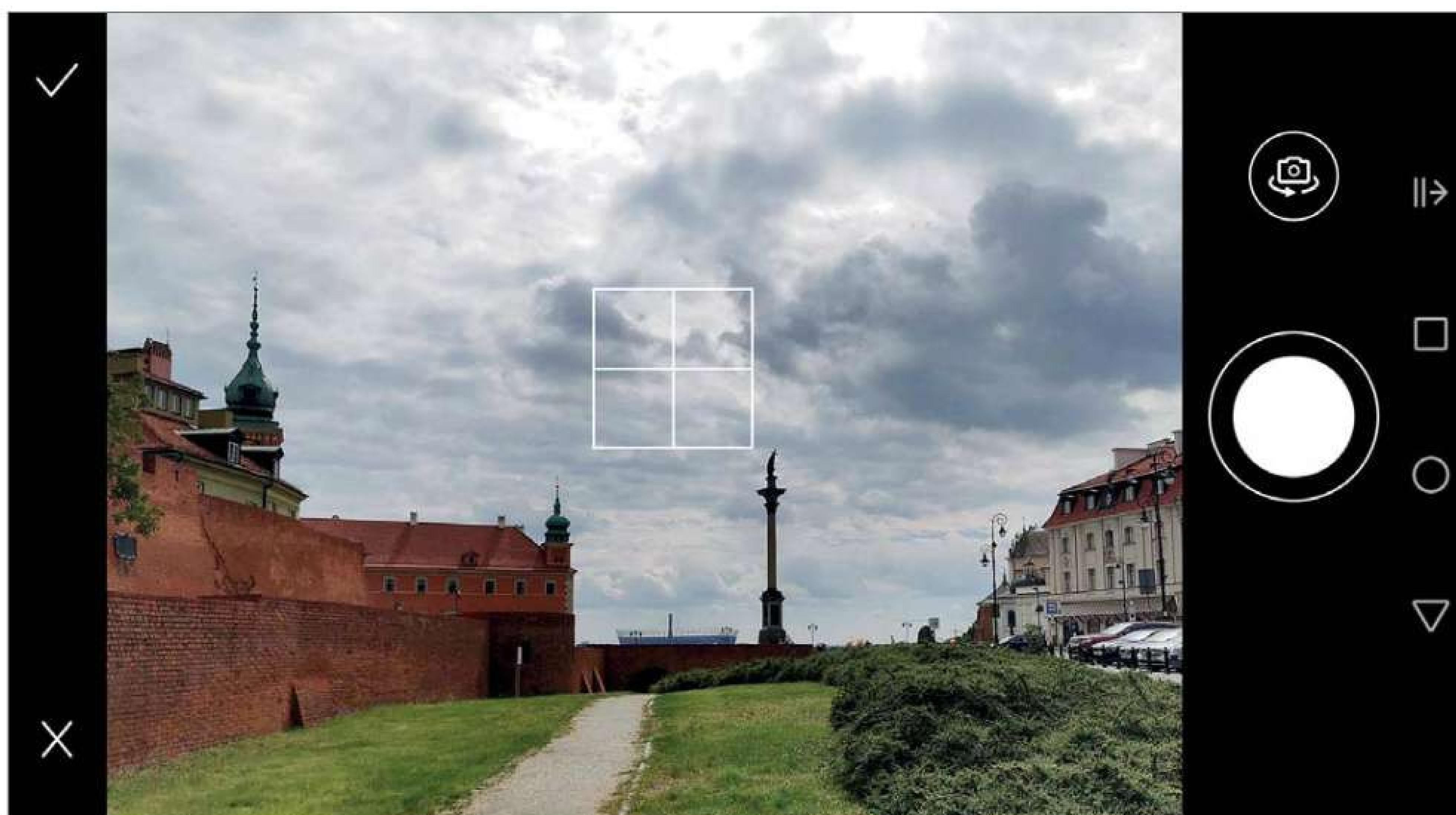
Rys. 4. Przykłady darmowych map dostępnych w aplikacji Traseo

Aby ruszyć w trasę, trzeba dotknąć przycisku . Od tej chwili każdy krok jest rejestrowany w aplikacji, a przebyta droga – kreślona na mapie. Aby robić zdjęcia, należy dotknąć ikony . Przy pierwszym użyciu po instalacji trzeba wyrazić zgodę na dostęp aplikacji do zdjęć i filmów.



Rys. 5. Pytanie o zezwolenie aplikacji na dostęp do zdjęć i filmów

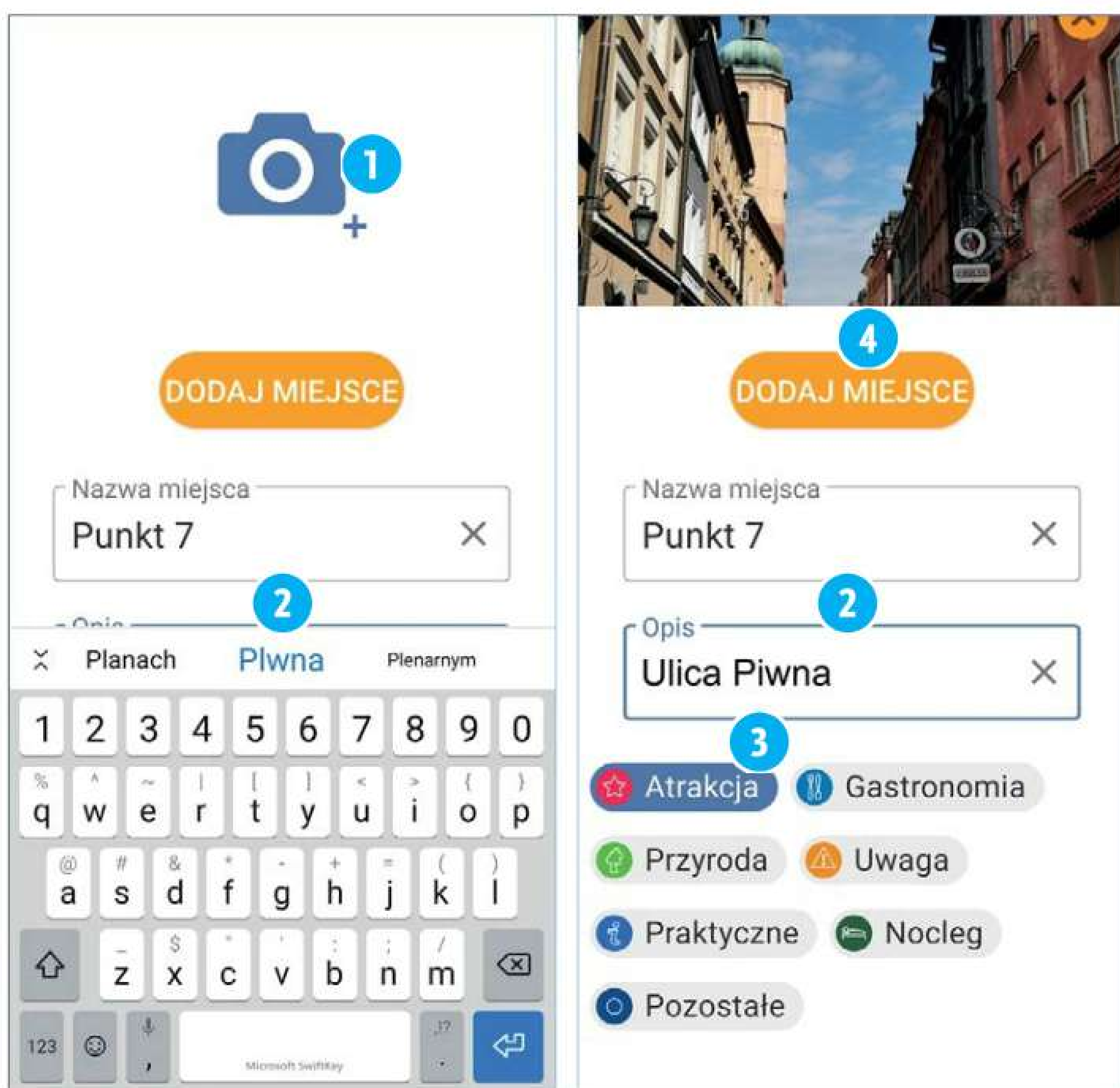
Potem wystarczy zrobić zdjęcie. Za pomocą przycisku ☑ akceptuje się umieszczenie zdjęcia na mapie, a za pomocą przycisku ☒ rezygnuje się z tej operacji.



Rys. 6. Robienie zdjęcia w aplikacji Traseo

Napotkany obiekt można też zaznaczyć na mapie – aby dodać do trasy punkt, trzeba dotknąć ikony (+), a następnie:

- dotknąć ikony aparatu (1), zrobić zdjęcie i zaakceptować jego umieszczenie;
- uzupełnić nazwę i opis punktu (2);
- wybrać kategorię obiektu (3);
- zapisać punkt (4) – na mapie pojawi się odpowiednia ikona.



Rys. 7. Zapisywanie punktu w aplikacji Traseo

PUBLIKOWANIE TRASY W SERWISIE TRASEO

Aby udostępnić wycieczkę innym użytkownikom aplikacji, należy zakończyć rejestrowanie trasy, tzn. dotknąć kolejno ⏸ i ⏹, następnie zapisać ją i wysłać do portalu **www.traseo.pl**.



Rys. 8. Zapisywanie trasy w aplikacji Traseo

Na portalu wystarczy nadać trasie nazwę (np. **Warszawska Starówka**), krótko ją opisać (np. **Atrakcje Starego Miasta**), określić rodzaj swojej aktywności (np. pieszo), wskazać miejsce zapisu (opcja **TELEFON & TRASEO.PL** umożliwia upublicznienie trasy, czyli udostępnienie jej wszystkim osobom korzystającym z aplikacji oraz portalu Traseo) i zapisać całość.

Aby edytować zapisaną na portalu trasę, należy zalogować się w serwisie za pomocą nazwy użytkownika i hasła przyjętych podczas tworzenia konta po zainstalowaniu aplikacji.



Rys. 9. Panel użytkownika na portalu **www.traseo.pl**

ZADANIA

1. Dobierzcie się w grupy dwu- lub trzyosobowe, tak aby w każdej grupie było jedno urządzenie mobilne z zainstalowaną aplikacją Traseo. Zaplanujcie pieszą wycieczkę po najbliższej okolicy, a następnie zarejestrujcie i opublikujcie jej trasę z zaznaczonymi atrakcjami. W razie potrzeby wprowadźcie stosowne zmiany trasy, edytując mapę na portalu.

2. Sprawdź, jakie inne aplikacje przeznaczone dla pieszych turystów są dostępne w internecie i jakie opcje oferują. Sporządź tabelę porównawczą dotyczącą trzech wybranych aplikacji.

5.5 Rozszerzona rzeczywistość

DOWIESZ SIĘ,

- co oznacza termin rozszerzona rzeczywistość (AR),
- gdzie wykorzystuje się technologię rozszerzonej rzeczywistości,
- jakie aplikacje AR mogą pomóc w obserwacjach astronomicznych.

Technologię rozszerzonej rzeczywistości zaczęto stosować w praktyce na początku lat 90. Obecnie jej ogromny potencjał jest efektywnie wykorzystywany w wielu dziedzinach, a ona wciąż się rozwija.

ROZSZERZONA RZECZYWISTOŚĆ – CO TO JEST?

Jedziesz samochodem do supermarketu. Jazda może być łatwiejsza, bo na przedniej szybie auta komputer pokładowy wyświetla informacje o warunkach drogowych i ostrzega przed możliwymi kolizjami oraz korkami na trasie. W sklepie nie potrzebujesz kartki z listą zakupów, ponieważ już masz ją przed oczami – wyświetlaną na szkłach okularów. Szybko idziesz w odpowiednie miejsca w poszukiwaniu potrzebnych produktów – w orientacji pomagają ci wskazówki ze smartfona skomunikowanego z okularami. Skaner wykrył, że batonik, który trzymasz w ręku, nie powinien się znaleźć w koszyku – zawiera orzeszki, na które masz uczulenie. Odkładasz go na regał i wybierasz inny, wskazany przez smartfon. Zauważasz na półce ładną porcelanową filiżankę – mrugasz, żeby zrobić jej zdjęcie swoimi okularami, i wysyłasz do mamy z pytaniem, czy kupić... Tak może wyglądać rozszerzona rzeczywistość w praktyce.

Rozszerzona rzeczywistość (**AR**, akronim od ang. *augmented reality*; czytaj: ogmentyd riality) to interaktywny system łączący świat realny z wirtualnym. Zazwyczaj wykorzystuje obraz z kamery, na który w czasie rzeczywistym są nakładane trójwymiarowa grafika oraz dźwięk.

Technologia VR

Technologia VR (akronim od ang. *virtual reality*; czytaj: werczułal riality) polega na kreowaniu otoczenia realnego (symulacje komputerowe) lub fikcyjnego (gry komputerowe). Rozszerzona rzeczywistość nie jest synonimem rzeczywistości wirtualnej.

Rozszerzoną rzeczywistość stosuje się w wielu dziedzinach, m.in.

- w grach – np. dzięki technologiom AR i GPS w Jurrasic World Alive gracz wyposażony w smartfon z kamerą „chwyta” dinozaury widoczne na tle rzeczywistego otoczenia;
- w systemach nawigacji – systemy oparte na AR nakładają informacje związane z trasą na to, co użytkownik widzi na ekranie;
- w galeriach, muzeach i innych tego typu placówkach – po nakierowaniu urządzenia mobilnego lub specjalnych okularów na eksponat na ekranie lub szkłach wyświetlają się dodatkowe materiały multimedialne na jego temat, np. ukazuje się świat generowany na podstawie treści obrazu, obiekt archeologiczny jest rekonstruowany;
- w edukacji – dzięki technologii AR można w atrakcyjny sposób rozpocząć przygodę z astronomią, wykonywać doświadczenia niemożliwe do zrealizowania w danych warunkach; studenci medycyny mogą przeprowadzać zabiegi na niby w warunkach podobnych do prawdziwych;
- w medycynie – zastosowanie technologii AR wspomaga obrazowanie, diagnostykę oraz zabiegi, np. specjaliści z warszawskiej Kliniki Kardiologii i Angiologii Interwencyjnej podczas zabiegu udrażniania tętnic wieńcowych używają reagujących na polecenia głosowe okularów Google Glass, dzięki którym mogą jednocześnie wykonywać operację i obserwować trójwymiarowe obrazy tomografii komputerowej serca pacjenta;
- w turystyce – technologię AR wykorzystuje m.in. aplikacja stworzona na zamówienie Gdańskiej Organizacji Turystycznej, prezentująca nieszablonowe lub nieoznakowane szlaki turystyczne;

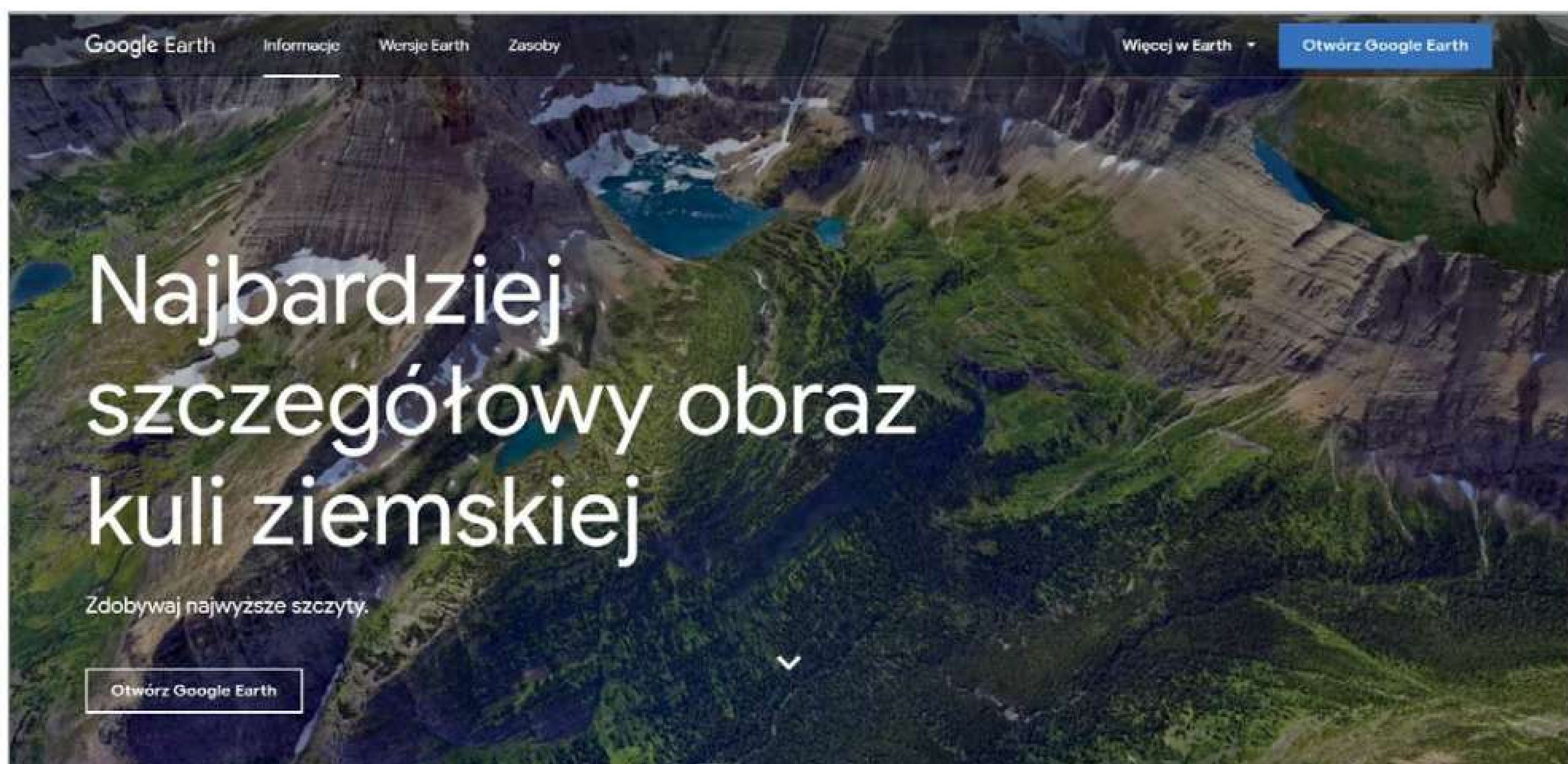


Rys. 1. Przykłady wykorzystania technologii rozszerzonej rzeczywistości

- w przemyśle odzieżowym, kosmetycznym i meblarskim – technologia AR pozwala stworzyć wirtualną przymierzalnię, która ułatwia zakupy online (na odbicie sylwetki lub twarzy użytkownika aplikacji są nakładane ubrania, akcesoria czy makijaż; na obraz rzeczywisty danego pomieszczenia są nanoszone meble, dzięki czemu można sprawdzić, czy się zmieszczą i czy będą się komponować z pozostałym wyposażeniem);
- w lotnictwie i motoryzacji – instrumenty pokładowe pokazują pilotom lub kierowcom dane na temat ukształtowania terenu, który widzą przed sobą lub którego nie widzą z powodu złych warunków atmosferycznych (poprawia się bezpieczeństwo jazdy, bo kierowca nie musi odrywać wzroku od drogi), aplikacja serwisowa podpowiada mechanikowi, jak naprawić konkretną część – pokazuje i omawia kolejne czynności, nakładając informacje na to, co mechanik aktualnie widzi.

GOOGLE EARTH

Program Google Earth (**google.com/earth/**, także aplikacja na urządzenia mobilne) ma bardzo rozbudowany system informacji oraz wizualizacji, dzięki czemu posiada też wiele cech rozszerzonej rzeczywistości i z biegiem czasu zyskuje ich coraz więcej.



Rys. 2. Fragment strony głównej Google Earth

W aplikacji Google Earth można korzystać z takich ciekawych narzędzi, jak:

- nawigacja 3D;
- wyszukiwanie miejscowości i planowanie tras;
- współpraca z odbiornikami GPS;
- baza hoteli, centrów rozrywki, restauracji, stacji benzynowych itp.;
- wyświetlanie granic administracyjnych i tras komunikacyjnych;
- wyświetlanie nazw miejscowości i obiektów geograficznych;
- pomiar odległości między zaznaczonymi punktami;

- panoramiczny widok ulicy;
- wyświetlanie modeli budynków w 3D (w tym zwiedzanie muzeów);
- symulator lotu;
- eksploracja nieba, kosmosu, Marsa i Księżyca;
- oglądanie przeszłości (wyświetlanie widoku danego miejsca sprzed lat – to, jak daleko można cofnąć się w czasie, zależy wyłącznie od istniejącego zbioru fotografii archiwalnych);
- wyprawy na dno oceanów;
- tryb „Odkrywca" (można odbyć przygotowane i wyreżyserowane wycieczki po wybranych miastach, budynkach oraz miejscach i zobaczyć je z nowej perspektywy);
- trójwymiarowe mapy.

► Zastanów się, które z tych narzędzi mają cechy AR.

KOSMOS W TWOIM TELEFONIE

Aby obserwować nocne niebo za pomocą telefonu lub tabletu, na który pobrano aplikację wykorzystującą technologię rozszerzonej rzeczywistości, trzeba aktywować geolokalizację oraz przyznać dostęp do kamery. Po skierowaniu kamery w niebo aplikacja skanuje widok, a następnie nakłada na obraz z kamery cyfrową mapę uwzględniającą położenie urządzenia – z obiektami znajdującymi się nad naszą głową.

Bezpłatna androidowa aplikacja Sky Map powstała w 2007 r. jako Google Sky Map, a pięć lat później jej kod został udostępniony na zasadzie *open source*. W menu można ograniczyć warstwy i wskazać elementy, które będą widoczne na podglądzie, np. gwiazdy, planety, horyzont. Wbudowana wyszukiwarka informuje, w którą stronę należy obrócić urządzenie, aby ujrzeć wybrany obiekt.



Rys. 3. Obraz z aplikacji Sky Map

W przypadku Star Walk 2 (pełne płatne wersje są dostępne dla systemów iOS i Android, okrojoną bezpłatną wersję aplikacji można pobrać na Androida) wystarczy stuknąć w wybrany obiekt na ekranie, aby wyświetliły się: nazwa, opis, galeria zdjęć wykonanych przez teleskopy oraz model 3D. Ciekawostkę stanowi funkcja obrazowania gwiazdozbiorów – po jej wybraniu na ekranie pokazują się stworzenia, rzeczy i postaci, od których gwiazdozbiory wzięły swoje nazwy.



Rys. 4. Obraz z aplikacji Star Walk 2

Stellarium to płatna aplikacja dla Androida oraz iOS, powiązana z dostępnym online programem StellariumWeb (**stellarium-web.org**). Pozwala m.in. dopasować stopień „zanieczyszczenia" nocnego nieba światłem oraz jasność – by lepiej było widać Drogę Mleczną. Wizualizacje gwiazdozbiorów mają charakter kulturowy, można wybrać ilustracje nawiązujące np. do kultury chińskiej, nordyckiej czy polinezyjskiej.

Wśród bezpłatnych i płatnych „kosmicznych" aplikacji wyróżniają się również:

- **Milky Way** – polska aplikacja przeznaczona do obserwacji nieba, dostępna dla użytkowników iPadów i iPhonów;
- **Satellite AR** – aplikacja pozwalająca śledzić lokalizację różnych satelitów, w tym Międzynarodowej Stacji Kosmicznej;
- **Our Universe AR** – po skierowaniu kamery na obraz wybranego ciała niebieskiego, wydrukowany wcześniej z serwisu międzynarodowego projektu Faulkes Telescope, i zeskanowaniu znacznika aplikacja prezentuje trójwymiarowy model planety bądź gwiazdy oraz podaje kluczowe informacje i ciekawostki na jej temat.

POJAZDY KOSMICZNE NA TWOIM BIURKU

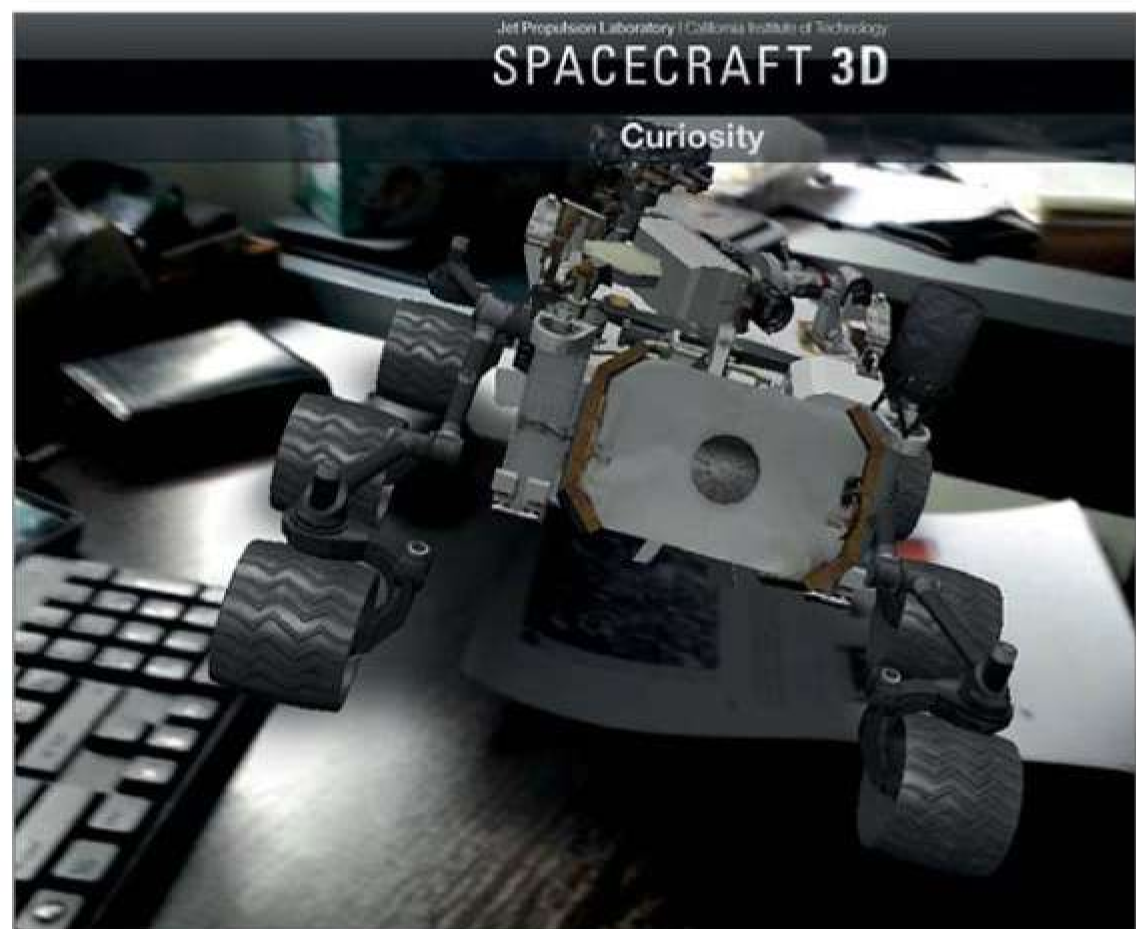
Dzięki technologii rozszerzonej rzeczywistości można też przyjrzeć się z bliska pojazdom kosmicznym używanym przez NASA w różnych programach kosmicznych. Bezpłatna anglojęzyczna aplikacja Spacecraft 3D, stworzona przez należące do NASA

centrum badawcze JPL, jest przeznaczona do urządzeń mobilnych z systemem Android (pod nazwą Spacecraft AR) lub iOS (Spacecraft 3D) i nie wymaga dostępu do internetu.

Aby na swoim biurku zobaczyć teleskop kosmiczny Hubble'a, marsjański łazik Curiosity czy opuszczającą Układ Słoneczny sondę Voyager, po zainstalowaniu aplikacji trzeba dodatkowo pobrać i wydrukować plik PDF z zestawem znaczników (**www.jpl.nasa.gov/apps/images/3dtarget.pdf**).

Aby zacząć zabawę, uruchom program Spacecraft AR (lub Spacecraft 3D), wybierz opcję **SELECT SPACECRAFT** (czytaj: selekt spejskraft; wybór pojazdu), a potem **Pick by marker** (czytaj: pik baj marker; wybór według znacznika). Gdy na ekranie pojawi się obraz z kamery, skieruj go na znacznik i przytrzymaj. Nad znacznikiem pojawi się twój pojazd. Niektórymi pojazdami, np. łazikiem marsjańskim Curiosity, możesz sterować (ikona z pojazdem i strzałką). Możesz także wysunąć antenę lub ramię wysięgnika. Po dotknięciu ikony z kółkiem zębatym wykonasz zdjęcie pojazdu, a po dotknięciu ikony z literą **i** – przeczytasz dotyczący go opis.





Rys. 5. Znacznik aplikacji Spacecraft 3D i zdjęcie z tej aplikacji – łazik marsjański na biurku

KOSMICZNY ROBOT W TWOIM DOMU

W filmach z serii *Gwiezdne wojny* bohaterom pomagają roboty. Oprócz humanoidalnego C3PO i beczkowatego R2D2 w *Przebudzeniu mocy* z 2015 r. pojawił się sferyczny robot BB-8. Rok później można go było kupić na Ziemi w wersji sterowanej za pomocą tabletu lub smartfona z systemem Android lub iOS.

BB-8 (Sphero iBall Star Wars w postaci kuli z doczepioną magnetycznie głową) może toczyć się po ustalonych wcześniej ścieżkach ruchu, patrolować teren, obracać się, świecić w różnych kolorach, tworzyć hologramy (dzięki wykorzystaniu technologii AR treści są wyświetlane na urządzeniu mobilnym użytkownika) oraz oglądać wspólnie z użytkownikiem filmy z serii i „spontanicznie" reagować na poszczególne sceny. Ale

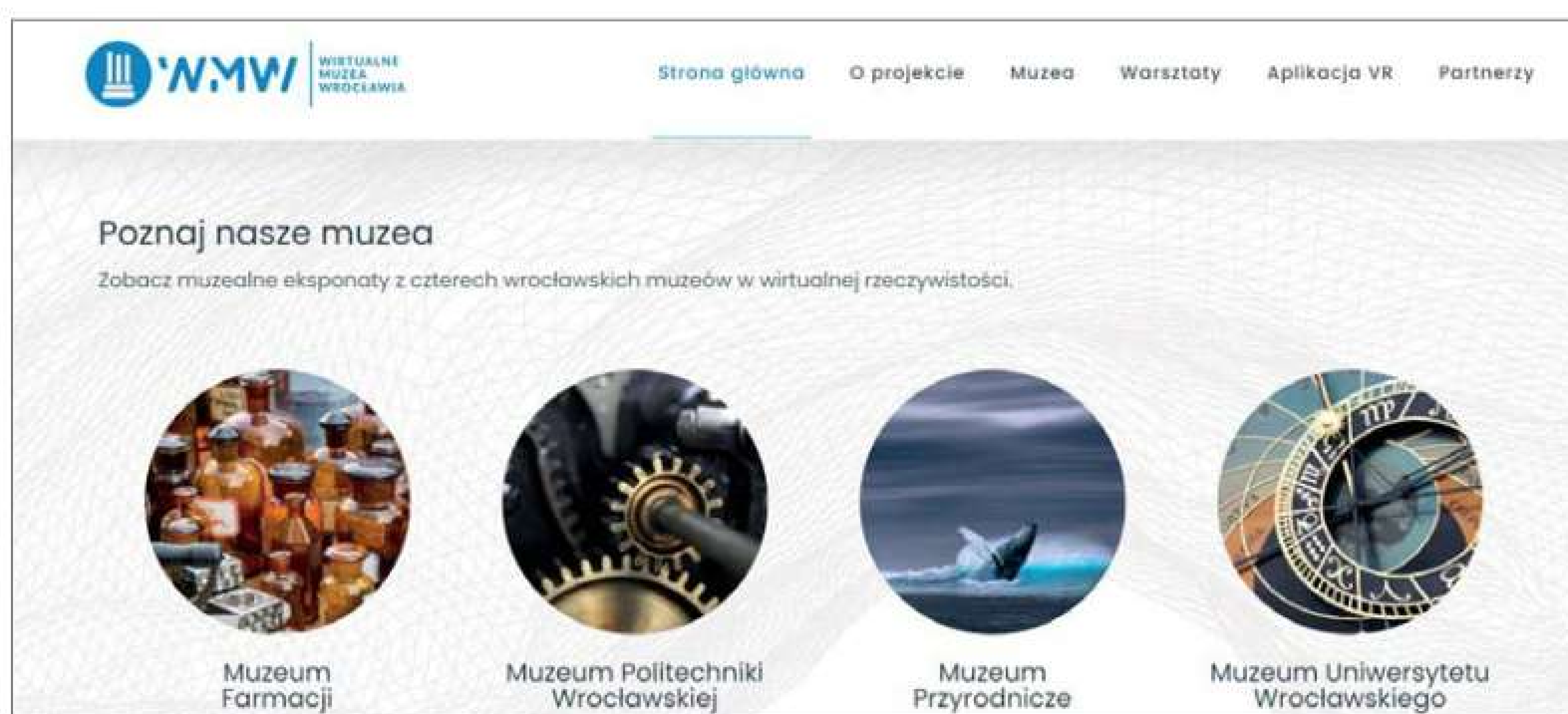
to nie wszystko! Osoby, które pobiorą aplikację Sphero Edu, mogą tworzyć programy sterujące robotem. Skrypty układa się za pomocą gotowych bloków: **Actions** (Akcje), **Controls** (Kontrola), **Operators** (Operatory), **Comparators** (Porównania), **Sensors** (Czujniki), **Events** (Zdarzenia), **Variables** (Zmienne), **Functions** (Funkcje). Gotowym programem można się podzielić z innymi użytkownikami.



Rys. 6. Sterowany za pomocą tabletu robot BB-8 oraz panel sterowania aplikacji Sphero Edu

ZADANIA

1. Sprawdź, czy w muzeum lub galerii w twojej okolicy nie ma wystawy posługującej się technologią AR lub VR. Jeśli tak, odwiedź ją i wykorzystaj odpowiednią aplikację podczas jej oglądania.



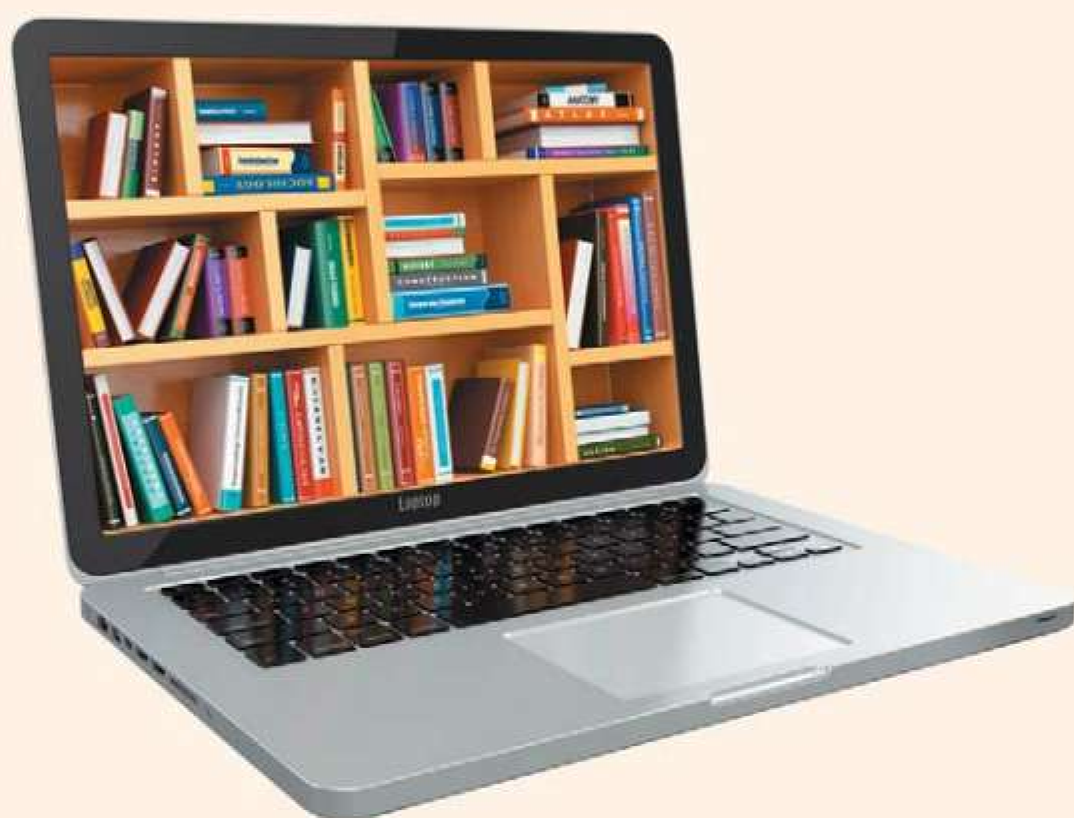
2. Zainstaluj wybraną aplikację do obserwacji nieba i zbadaj jej możliwości. Przeprowadź obserwacje gwiazdozbiorów na niewidocznej u nas południowej półkuli. Opisz wyniki w dokumencie edytora tekstu.

3. Skorzystaj z aplikacji Spacecraft AR (Spacecraft 3D), by zrobić zdjęcia różnych pojazdów kosmicznych, a następnie utwórz z nich kolaż.

Jak rozwijać zainteresowania online?

W sieci znajdziesz bardzo dużo ciekawych zasobów. Serwisami **www.pl.euhou.net** i **www.galaxyzoo.org/?lang=pl** powinni zainteresować się amatorzy astronomii. W serwisie **www.ted.com** można wyszukać rozmaite interesujące filmy z wykładami dotyczącymi m.in. technologii, nauki i kultury. Warto zajrzeć również na platformę filmów edukacyjnych **ed.ted.com**, która stanowi rozwinięcie serwisu TED.





Jak internet pomaga w samodzielnej nauce?

Warto pamiętać, że różne instytucje edukacyjne oferują ciekawe szkolenia online, nierzadko bezpłatnie. Na przykład w serwisie Akademii Khana, która współpracuje z takimi prestiżowymi partnerami jak NASA, MoMA (ang. *Museum of Modern Art* – Muzeum Sztuki Nowoczesnej) i MIT (ang. *Massachusetts Institute of Technology* – Instytut Technologii w Massachusetts), znajdziesz ćwiczenia oraz filmy instruktażowe z zakresu nauk ścisłych, przyrodniczych oraz humanistycznych.

Czym się różnią rzeczywistość wirtualna i rzeczywistość rozszerzona?

Rzeczywistość wirtualna (VR) oznacza symulację wymyślonego otoczenia, najczęściej oglądanego na ekranie smartfona za pomocą specjalnych okularów. Rzeczywistość rozszerzona (AR) to interaktywny system łączący światy realny i wirtualny, zwykle wykorzystujący obraz z kamery, na który są nakładane grafika 3D i dźwięk tworzone w czasie rzeczywistym. Dzięki programom rozszerzonej rzeczywistości możesz zdobyć wiedzę nie tylko ciekawą, ale też nieosiągalną w inny sposób.



H

I

J

K

L

M

N

O

P

S

T

U

W

Źródła ilustracji

Okładka: (robot) kirill_makarov/Shutterstock.com.

Strony działowe: s. 5 (chłopak z laptopem) pikselstock/Shutterstock.com, (tło) Vit-Mar/Shutterstock.com; s. 49 (logo Pythona na tle kodu) Trismegist san/Shutterstock.com, (dziewczyna z laptopem) Halfpoint/Shutterstock.com; s. 121 (dziewczyna z mapą w górach) Daniel Jenny/Shutterstock.com, (kości do gry) Jalin/Shutterstock.com; s. 159 (przykład wykorzystania technologii AR) panuwat phimpha/Shutterstock.com, (chłopak z tabletem) MvanCaspel/Shutterstock.com.

Podsumowania: s. 48 (symboliczna prezentacja CSS) BSVIT/Shutterstock.com, (buttony) newart-graphics/Shutterstock.com, (wskaźnik) Tartila/Shutterstock.com; s. 84 (laptop i pyton) Prokradyha/Shutterstock.com, (kolorowa pętla) Marish/Shutterstock.com, (strumień zer i jedynek) USBFCO/Shutterstock.com; s. 120 (symboliczne ujęcie pracy w arkuszu kalkulacyjnym) Nakigitsune-sama/Shutterstock.com, (wykresy) bsd/Shutterstock.com, (symboliczne ujęcie pracy laptopa) klyaksun/Shutterstock.com; s. 158 (symboliczne ujęcie symulacji) Tartila/Shutterstock.com, (fraktal) Stocklady/Shutterstock.com (aplikacja mapowa) Aleksandr_a/Shutterstock.com; s. 193 (symboliczne ujęcie rozwijania zainteresowań online) Incomible/Shutterstock.com, (symboliczne ujęcie biblioteki cyfrowej) Maxx-Studio/Shutterstock.com, (gogle VR) Anton Gvozdikov/Shutterstock.com.

Infografika: s. 7 (żyrafa) posscriptum/Shutterstock.com.

Tekst główny: s. 21 (schemat mieszania kolorów RGB) MicroOne/Shutterstock.com; s. 23 (sójka i kopciuszek) Mirosław Wyczółkowski; s. 30 (foki) Mirosław Wyczółkowski; s. 32 (kawka) Mirosław Wyczółkowski; s. 34 (widok z drzewem na pierwszym planie) Mirosław Wyczółkowski; s. 35 (galeria prezentująca ptaki) Mirosław Wyczółkowski; s. 37 i 41 (lew i galeria zwierząt *Moje małe zoo*) Mirosław Wyczółkowski; s. 74 (trasa między kontynentami) artyway/Shutterstock.com, (samolot) phive/Shutterstock.com; s. 85 (monitor) Rawpixel.com/Shutterstock.com, (wykres funkcji liniowych) IconBunny/Shutterstock.com; s. 124 (dwa króliki) Litvalifa/Shutterstock.com; s. 127 (złoty podział) upstock/Shutterstock.com; s. 134 (kości) Mine Eyes Design/Shutterstock.com; s. 135 (aloes) Sabine Hortebusch/Shutterstock.com, (klatka schodowa) ledmark/Shutterstock.com, (elementy samochodu) Mikhail Leonov/Shutterstock.com; s. 141 (płatki śniegu pod mikroskopem) ch123/Shutterstock.com; s. 148 (płytka micro:bit z przodu) urbanbuzz/Shutterstock.com; s. 149 (płytka micro:bit z tyłu) urbanbuzz/Shutterstock.com; s. 150 (doświadczenie przeprowadzane z wykorzystaniem płytki micro:bit) Witold Kranas; s. 187 (komórka z aplikacją AR umożliwiającą prezentację rysunku w wersji 3D) Zyabich/Shutterstock.com, (tablet z informacjami na temat Bramy Brandenburskiej) Andrey_Popov/Shutterstock.com, (komórka z wirtualną przymierzalnią mebli) Zapp2Photo/Shutterstock.com, (naprawa samochodu z wykorzystaniem AR) Zapp2Photo/Shutterstock.com; s. 192 (robot sterowany za pomocą tabletu) Witold Kranas.

INFORMATYKA

O podręczniku



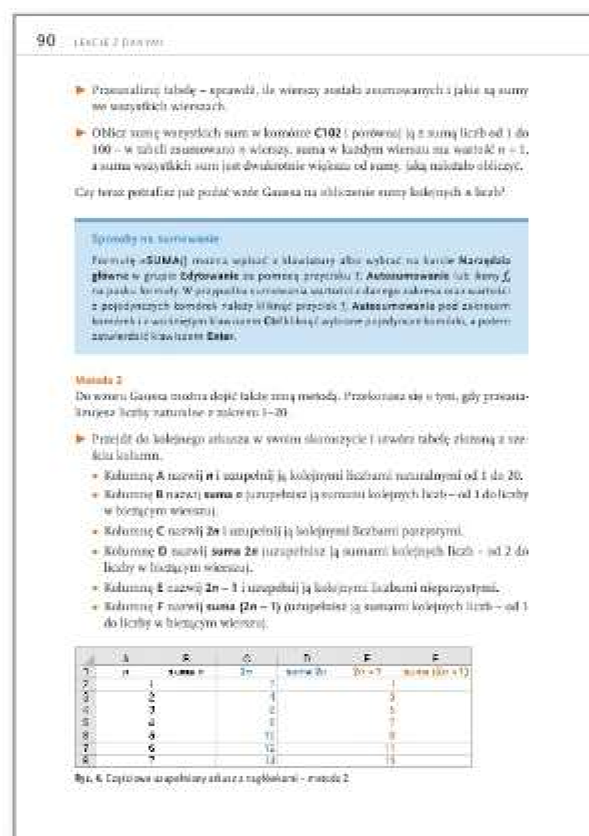
Strona działowa
Krótkie wprowadzenie w tematykę rozdziału.



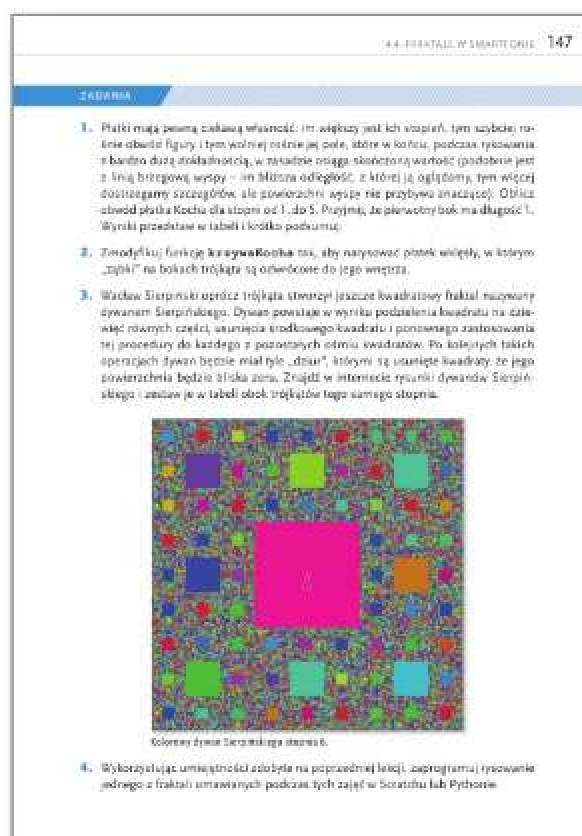
Lista zagadnień
Omawiane tematy i kluczowe umiejętności.



Zrzuty ekranowe
Pomogą zapoznać się z opcjami programów.



Wskazówki
Ułatwią zapamiętanie ważnych zasad.



Zadania
Pomogą sprawdzić wiedzę i umiejętności.



Podsumowanie
Skrót wiadomości z danego rozdziału.



WYDAWNICTWA SZKOLNE i PEDAGOGICZNE

wsip.pl
sklep.wsip.pl
infolinia: 801 220 555

ProfiLingua
profi-lingua.pl